1985

An adaptive model based disk file head positioning servo system.

Wikstrom, Kenneth R.

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

AN ADAPTIVE MODEL BASED DISK FILE
HEAD POSITIONING SERVO SYSTEM

by

Kenneth R. Wikstrom

September 1985

Thesis Advisor: G. J. Thaler

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Adaptive Model Based Disk File Head Positioning Servo System		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1985
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kenneth R. Wikstrom		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943-5100		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, Caliofornia 93943-5100		12. REPORT DATE September 1985
		13. NUMBER OF PAGES 146
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release;distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Adaptive model
Disk file head positioning servo

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The feasibility of controlling a disk file head positioning servo with an adaptive computer simulation model is investigated. During the seek mode, model updating of position, velocity, and motor gain parameter are accomplished from samples of head position only, thereby eliminating the requirement for a tachometer in the positioning servo system. In the track follow mode, three methods of linear compensation are presented to settle the transients of

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20.
the servo and allow for the positioning of the read/write head on track center in minimum time. Implementation of the simulation model into a microprocessor is also investigated.

An Adaptive Model Based Disk File Head Positioning Servo System

by

Kenneth R. Wikstrom
Lieutenant, United States Navy
B.S.E.E., Purdue University, 1979

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1985

ABSTRACT

The feasibility of controlling a disk file head positioning servo with an adaptive computer simulation model is investigated. During the seek mode, model updating of position, velocity, and motor gain parameter are accomplished from samples of head position only, thereby eliminating the requirement for a tachometer in the positioning servo system. In the track follow mode, three methods of linear compensation are presented to settle the transients of the servo and allow for the positioning of the read/write head on track center in minimum time. Implementation of the simulation model into a microprocessor is also investigated.

TABLE OF CONTENTS

I. INTRODUCTION

As the demand for digital data storage has increased, technological advances in the disk file industry have led to higher data track densities on rotating magnetic disk files. Track densities of 1000 tracks per inch (TPI) or greater are not uncommon in today's market. Because the corresponding track widths are often .001" or less, precise means of positioning a read/write head over a given data track must be developed. This thesis will examine the feasibility of controlling a disk file head positioning servo through the use of a microprocessor-based adaptive model and compensation scheme.

Figure 1.1 is a sketch of the system to be investigated. The magnetic disk rotates at a constant speed w(t). When the desired track number command is received from the main computer CPU, the microprocessor generates an appropriate servo motor drive signal to cause the servo motor to rotate through an angle O(t). This rotation, acting through the actuator arm of length L, causes the head to translate through a circular arc that is approximately along the radius of the rotating disk. Because the data tracks are arranged in concentric circles about the center of the disk, this results in the movement of the head across one or more

tracks until the desired track is found.

The microprocessor must perform two tasks to accomplish the move to the desired track:

1. It must generate motor drive signals to move the head from the present track to any other commanded track in minimum time. This is called the seek mode of the operation.
2. It must center the head over the desired track and maintain this position on the disk surface while the head is reading from or writing onto the data track. This is called the track follow mode of operation.

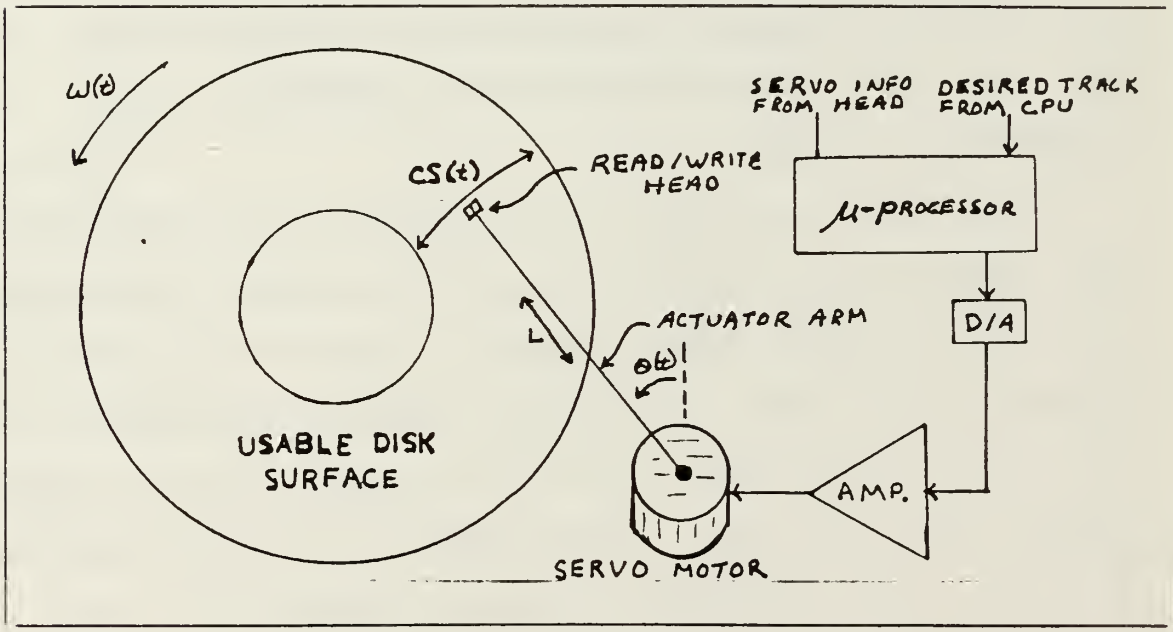

Figure 1.1 Sketch of Disk File Head Positioning Servo System

To accomplish the seek function, the microprocessor will utilize an adaptive model to simulate the actions of the

amplifier, servo motor and actuator arm as they move the head across the radius of the disk surface. The actual servo motor is to be driven open loop thus eliminating the requirement for a tachometer to determine the velocity of the head.

In the track follow mode, the microprocessor provides linear compensation to the servo motor. The compensation settles the transients of the motor and allows the head to "ride" the center of the data track while it performs its read/write functions.

Both functions of the microprocessor require accurate measurements of head position as it moves across the disk surface or attempts to follow the desired data track. One scheme to provide the position information is to imbed the servo information into the data track itself. Bursts of servo information consisting of a track number code, commonly a GREY code, and an error code identify the track and generate an error signal proportional to distance and direction from track center. Figure 1.2 illustrates the layout of these narrow sectors of servo information within the data tracks on the disk surface. Since this information is provided in sectors equally spaced around the disk, position information is available only at discrete instants of time as the rotating sectors pass under the head. Thus the adaptive model and compensation loop must operate as a sampled data servo system and the head has the additional

responsibility of providing these samples of head position to the microprocessor.

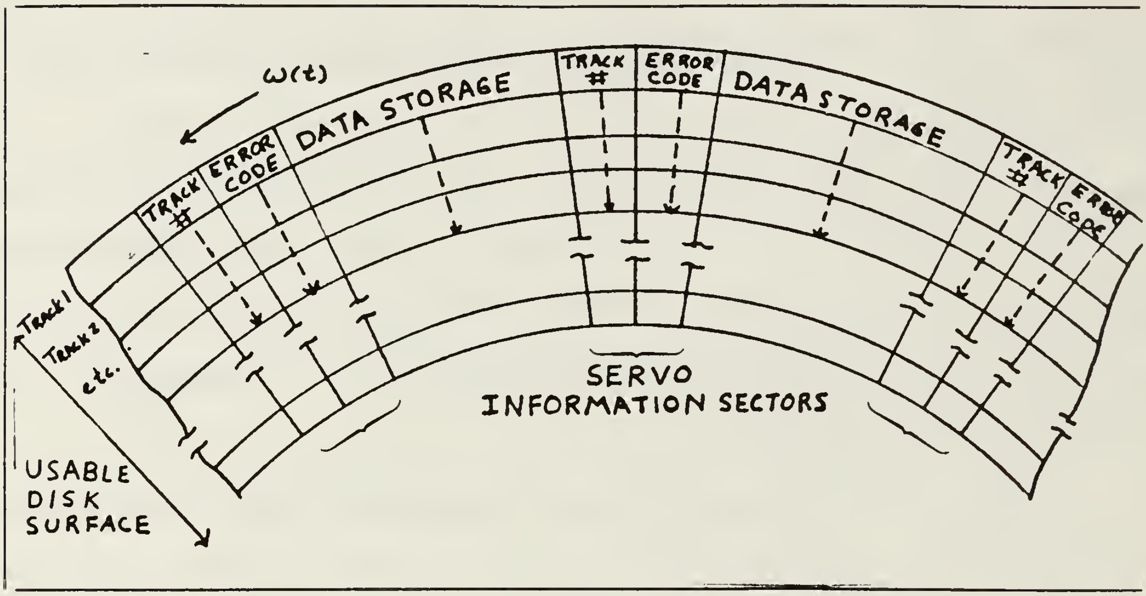

Figure 1.2 Proposed Layout of Track Information

Chapters 2 and 3 will treat the development of the adaptive simulation model and present simulation studies of the performance of this adaptive model scheme for the seek mode of operation. Testing of the adaptive model continues in Chapter 4 showing the effects of varying amplifier parameters and changing the servo motor transfer function. Chapter 5 discusses the development of a current source drive system and presents simulation studies to show how this type of drive can be used to overcome the effects of servo motor parameters. Three methods of providing linear compensation to the servo motor during the track follow mode are presented in Chapter 6. The effects of varying the

sampling period and introducing time delays in the system are studied in Chapter 7.

Finally, Chapters 8 and 9 will discuss the implementation of the complete system into a microprocessor and areas for further study. Appendices A through D are provided to list the DSL/VS simulation programs and appropriate simulation data printouts used in the course of this thesis research.

II. DEVELOPMENT OF THE SIMULATION MODEL

A. INTRODUCTION

The simulation model chosen was a servo with a curve following velocity loop as shown in Figure 2.1. This model operates in two modes for a step position command:

1. An initial full acceleration mode
2. A curve following mode

When the curve to be followed is chosen to be the deceleration curve for the idealized motor, the model will be a practical application of bang-bang control [Ref. 1].

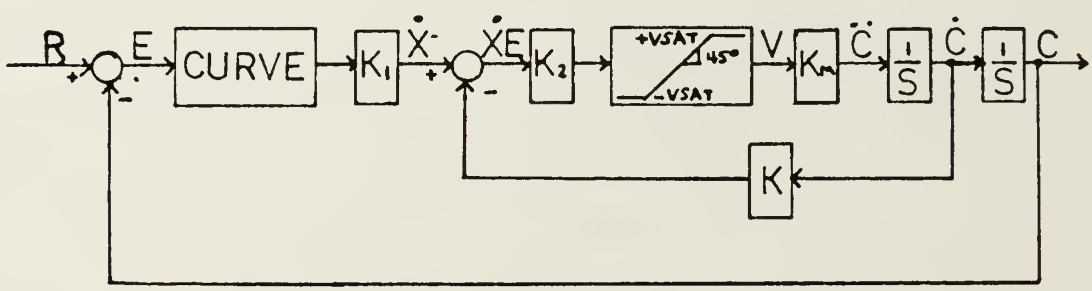

Figure 2.1 Block Diagram of the Simulation Model

When a step input is applied to the model, the error signal (E) will enter the curve and produce a commanded

velocity input ($\dot{X}$) to the velocity loop. The amplifier saturates and full forward drive signal is applied to the motor (full acceleration mode). As the position error signal decreases, the commanded velocity is reduced until it is equal to the velocity feedback signal ($K\dot{C}$). Because the commanded velocity signal is decreasing, the velocity error ($\dot{X}E$) will go negative and cause the voltage signal to the motor to reverse. By proper selection of the amplifier gain, the system will continue to switch between the positive and negative saturation limits of the amplifier and follow the curve down until the desired position is reached.

B. CURVE DESIGN

To enable the system to follow the curve to the desired position, the curve must be chosen carefully. A parabolic curve was used because it approximates the deceleration curve of an ideal motor. Experience gained through simulation studies has shown that the curve should be below the motor deceleration curve to enable the system to accurately track it. To accomplish this, the gain constant K1 was set to a value of 0.8.

The equation of the curve was derived from the idealized motor equations as follows

$$\ddot{C} = Km\ Vsat \tag{2.1}$$

$$\dot{C} = \int \ddot{C}\ dt = Km\ Vsat\ t + \dot{C}(0),\quad (\dot{C}(0)=0) \tag{2.2}$$

$$C = \int \dot{C}\ dt = \frac{Km\ Vsat\ t^2}{2} + C(0) \tag{2.3}$$

From equation 2.2,

$$t = \dot{C}/Km\ Vsat \tag{2.4}$$

and substituting into equation 2.3,

$$C = \frac{Km\ Vsat}{2} \cdot \left(\frac{\dot{C}}{Km\ Vsat}\right)^2 = \frac{\dot{C}^2}{2\ Km\ Vsat} \tag{2.5}$$

For deceleration from initial conditions with the input R = 0,

$$C = -E \tag{2.6}$$

$$\dot{C} = -\dot{E} \tag{2.7}$$

$$E = \frac{\dot{E}^2}{2\ Km\ Vsat} \tag{2.8}$$

$$\dot{E} = \sqrt{2\ Km\ Vsat} \qquad \sqrt{E} \tag{2.9}$$

Letting

$$A = \sqrt{2\ Km\ Vsat} \tag{2.10}$$

and

$$\dot{X} = \dot{E} \tag{2.11}$$

$$\dot{X} = A\sqrt{E} = \text{commanded velocity} \tag{2.12}$$

Thus, the commanded veolocity curve can be generated by an initial calculation of the parameter A, then continuously multiplying this factor by the square root of the error signal. It can also be generated by precalculating the curve and storing the values in a table look up memory yielding commanded velocity values for discrete values of the error signals.

C. OTHER MODEL PARAMETERS

The motor gain constant (Km) is determined by the transfer function of the motor that the simulation model will control. The value of Km chosen was 300.0 (to be derived in Chapter 3).

The saturation limits $\pm$ Vsat of the saturating amplifier are determined by available power supply voltages. This limit was arbitrarily chosen as $\pm$ 10.0 volts. The gain parameter (K2) of the linear portion of the amplifier is chosen to saturate the amplifier when the step position command is one or more tracks. Assuming a track density of 1000 tracks per inch radially, for a one track step input:

$$\dot{X} \cdot K2 = K1 \cdot A \sqrt{E} \cdot K2 = (0.8) \sqrt{2 \cdot 300 \cdot 10} \sqrt{.001 \cdot K}$$

$$= 2 \cdot K2 \geq Vsat = 10$$

Thus, the value of K2 must be greater than 5.0. The value of K2 will also effect the bandwidth of the linear track follow mode of operation (to be discussed in Chapter 6). As a result, K2 was chosen to be 10,000 to effectively allow the amplifier to act as a switch to provide full forward and reverse drive signals to the motor in the curve following mode.

The gain of the velocity feedback channel (K) is chosen such that $\dot{X} = K \dot{C}$ when the simulation motor velocity ($\dot{C}$) is at the desired speed for a given step position input. Using the same arguments in the derivation of the parameter A, for deceleration from initial conditions with R = 0

$$-C = E \qquad (2.6)$$

$$-\dot{C} = \dot{E} \qquad (2.7)$$

$$\dot{X} = K1 \cdot \dot{E} = -K\,\dot{C} = K\,\dot{E} \qquad (2.13)$$

Therefore, the parameter K should be equal to K1. Again from simulation studies, close curve following was achieved when K is set to unity (1.0).

D. SIMULATION STUDIES OF THE BASIC MODEL

To demonstrate the curve following ability of this scheme, the model was simulated using DSL/VS. Appendix A lists the DSL simulation program used in this study. All parameters and variables used in the program are as previously discussed in this chapter and the signal flow follows the block diagram of Figure 2.1. The phase plane plot (velocity $\dot{C}$ versus position C) is shown in Figure 2.2 for a step position command of 0.1 inch (100 tracks). The figure shows the velocity of the model motor increases until it crosses the commanded velocity curve ($\dot{X}$) then follows this curve to the commanded position. Figure 2.3 shows the step response for the model during the seek mode. Figures 2.4 and 2.5 are the phase plane plot and step response for a step position command of 0.001 inch (1 track). The figures show that the model performs well even for the track to track move.

Now that a working simulation model has been found, use of this model to control the seek mode of the disk file head positioning servo will be investigated.

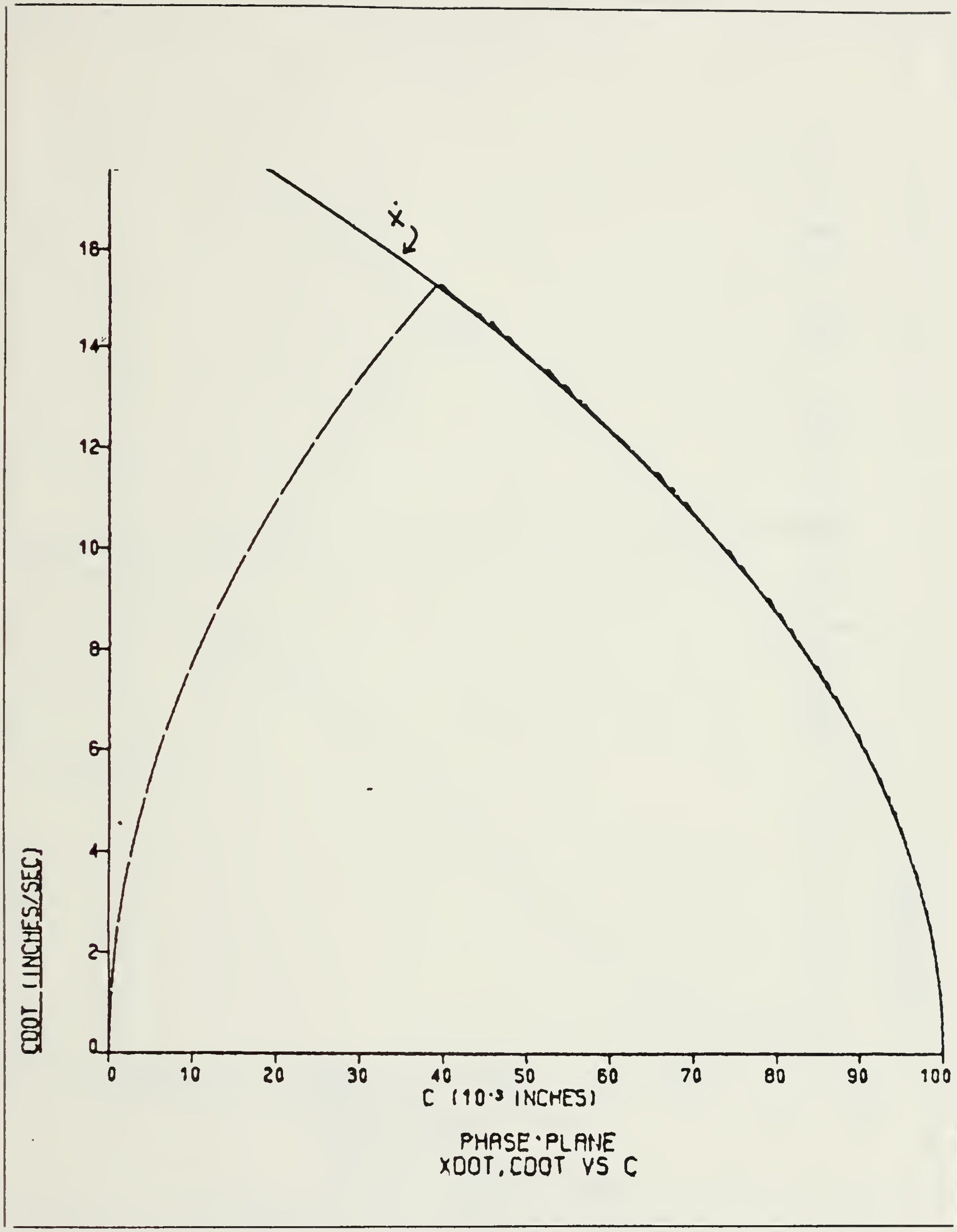

Figure 2.2 Phase Plane Trajectory of the Model
(100 Track Move)

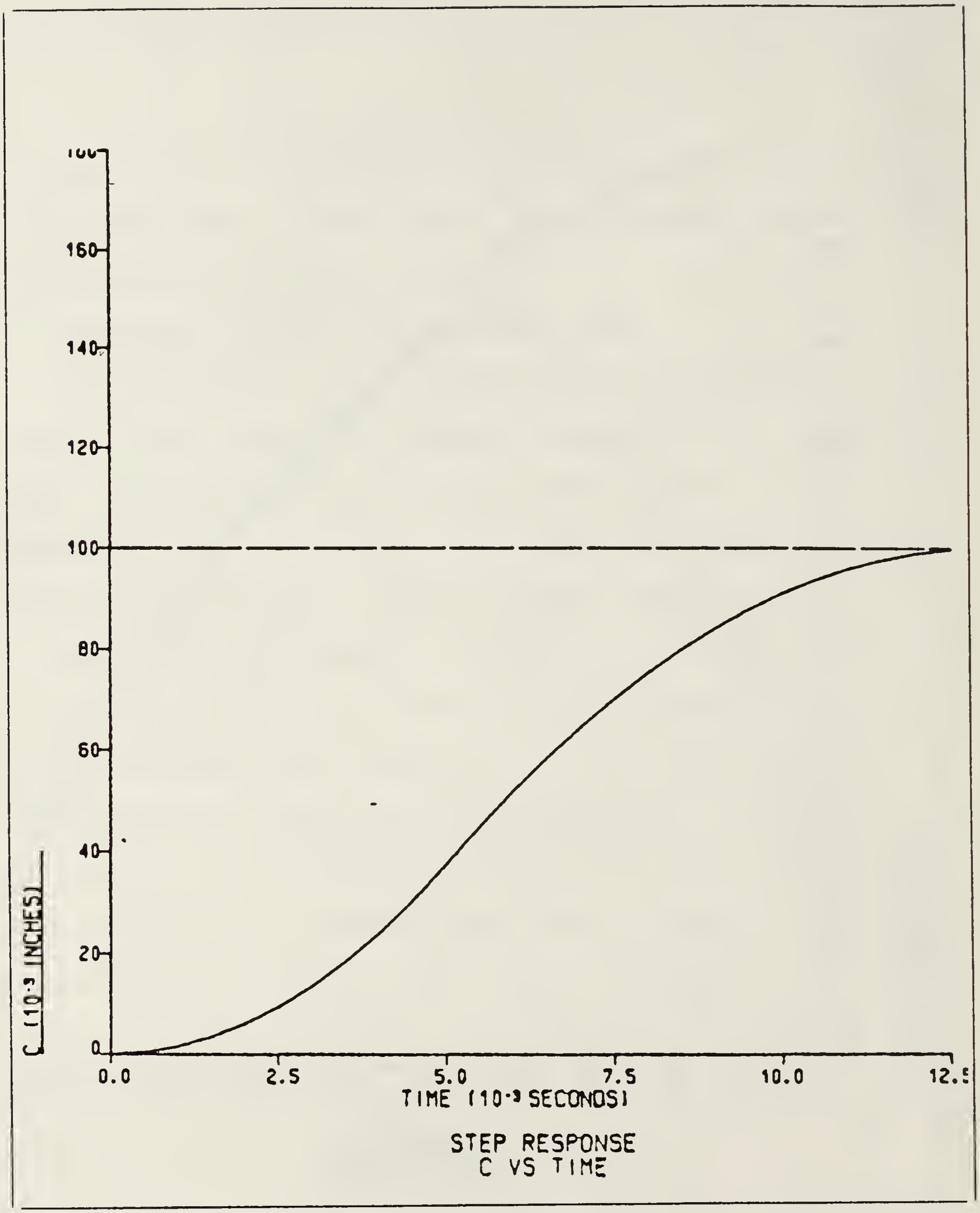

Figure 2.3 Step Response of the Model (100 Track Move)

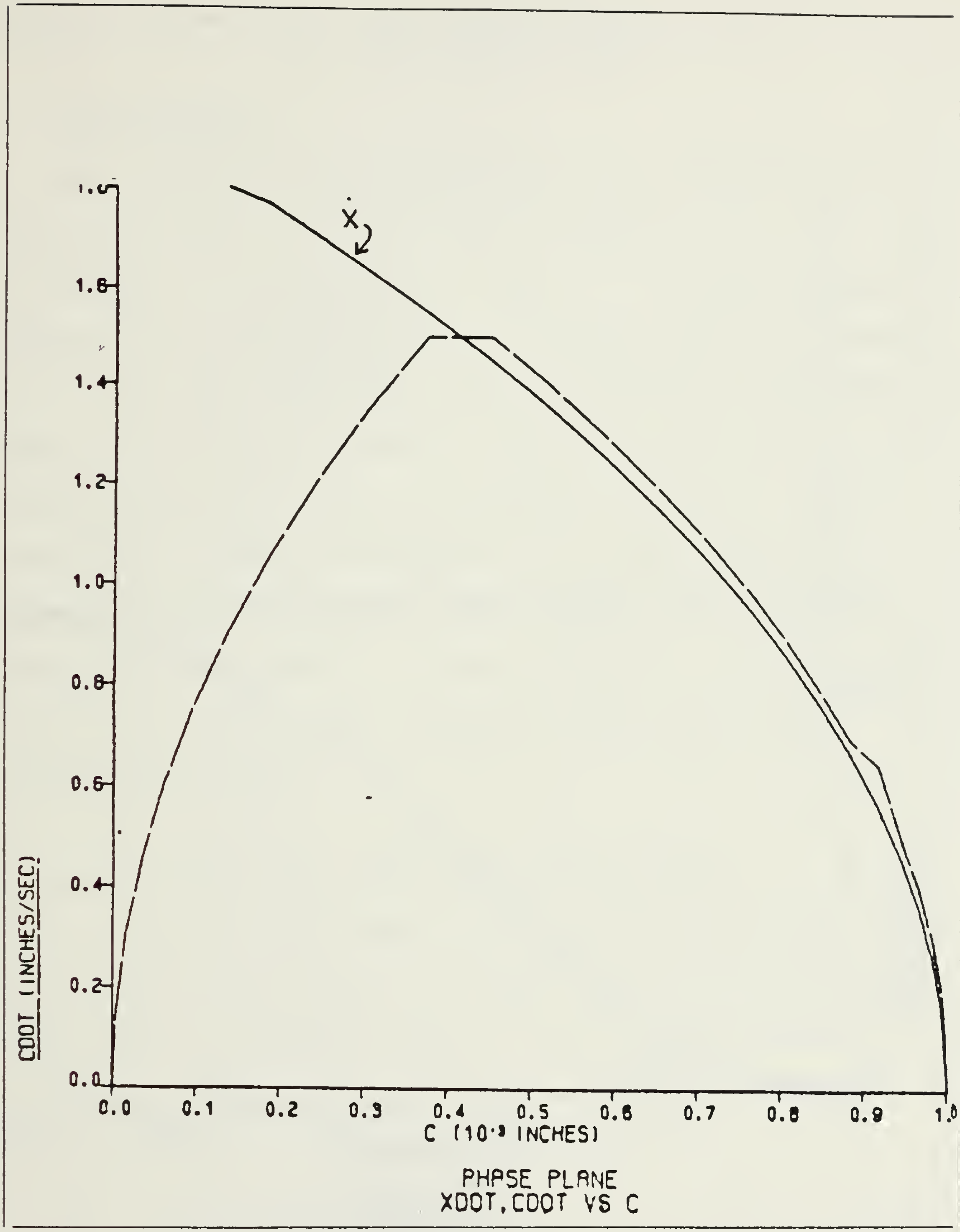

Figure 2.4 Phase Plane Trajectory of the Model
(1 Track Move)

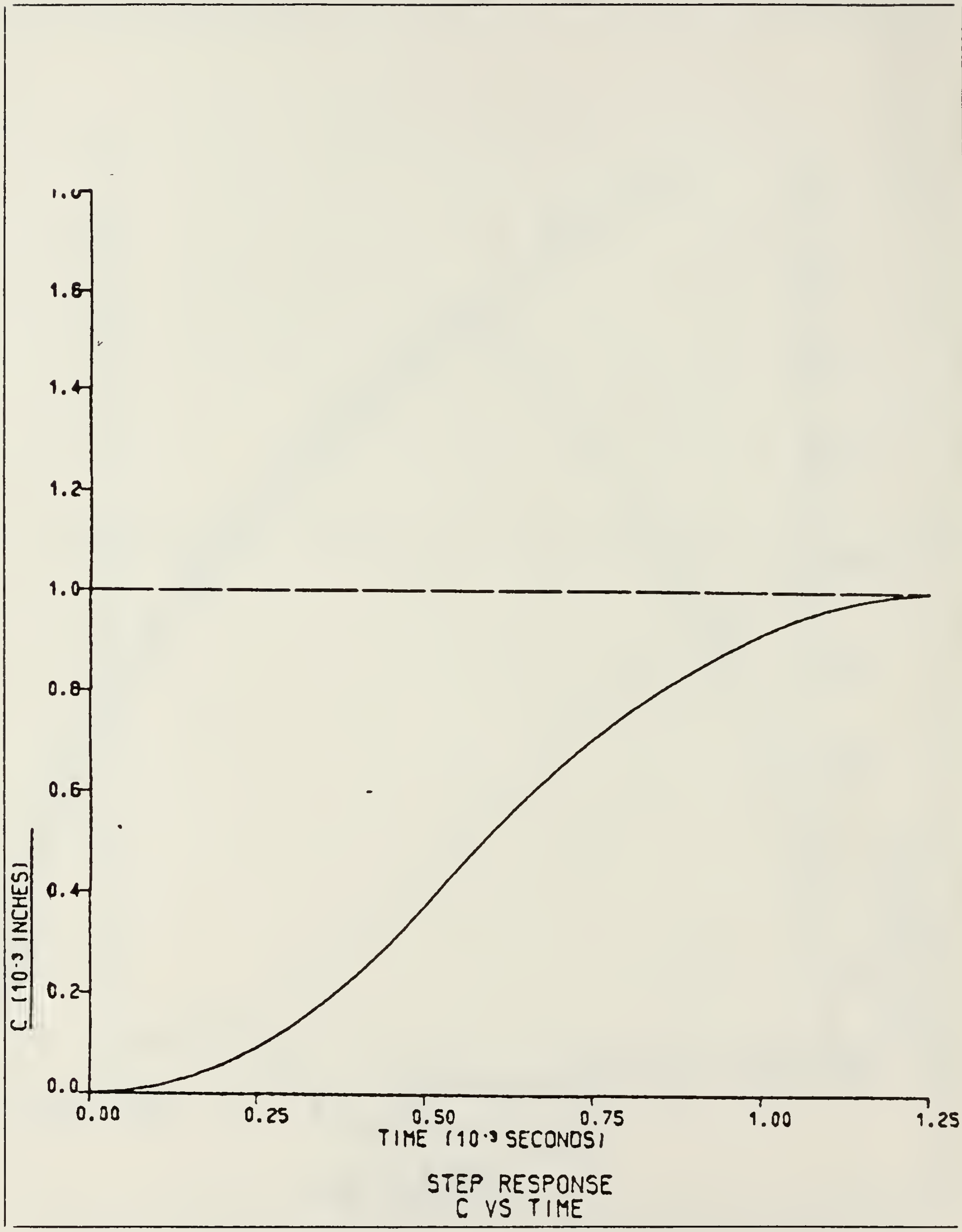

Figure 2.5 Step Response of the Model (1 Track Move)

III. THE ADAPTIVE MODEL

A. INTRODUCTION

The microprocessor simulation model of Chapter 2 is to be used to drive a positioning servo open loop so as to make it follow a predetermined curve to a desired position command. To accomplish this, the model must be adaptive, i.e., it must "mimic" the states (position, velocity) and have a motor gain constant equivalent to the non-ideal positioning servo. This discussion will first describe the positioning servo selected for use in the testing of the adaptive scheme and will then present the algorithm to update the model states and gain parameter in the seek mode. Simulation studies of the adaptive model are included at the end of this chapter.

B. POSITIONING SERVO MOTOR SELECTED FOR TEST

To realistically test the ability of the adaptive model to drive a positioning servo open loop, a permanent magnet DC servo motor currently used in the disk file industry was selected. Through analysis of experimental test data, the transfer function of the servo was found

$$G(s) = \frac{K}{s(s + Pm)(s + Pe)} \tag{3.1}$$

where

$K = 9.68 \times 10^{5}$ radians/volt

Pm = mechanical pole = 20.55 radians/sec

Pe = electrical pole = 3536.0 radians/sec

Rewriting in Bode form

$$G(s) = \frac{13.3}{s(s/Pm + 1)(s/Pe + 1)} \quad \text{radians/volts} \tag{3.2}$$

The open loop Bode plot for the servo motor is shown in Figure 3.1. The open loop Bode plot with the pole at Pe removed from equation 3.2 is shown in Figure 3.2. Comparison of the two plots showed no significant change in the magnitude response curves by removing the pole at Pe (ignoring the electrical pole) from the transfer function.

To convert the rotational motion of the servo motor to a translation across the radius of a disk file, an actuator arm length of 1.0 inches was used. Although this is unreasonably small, the arm length can be adjusted to suit a particular hardware design when implemented into an actual system. The resulting transfer function used for the simulation of the servo motor is shown below:

$$G(s) = \frac{13.3}{s(s/20.55 + 1)} \quad \text{inches/volt} \tag{3.3}$$

$$= \frac{CS(s)}{VS(s)}$$

where

$CS(s)$ = servo motor position

$VS(s)$ = servo motor drive voltage

An additional open loop Bode Plot was made for a Km/s motor that is asymptotic to the -40dB/decade slope of the

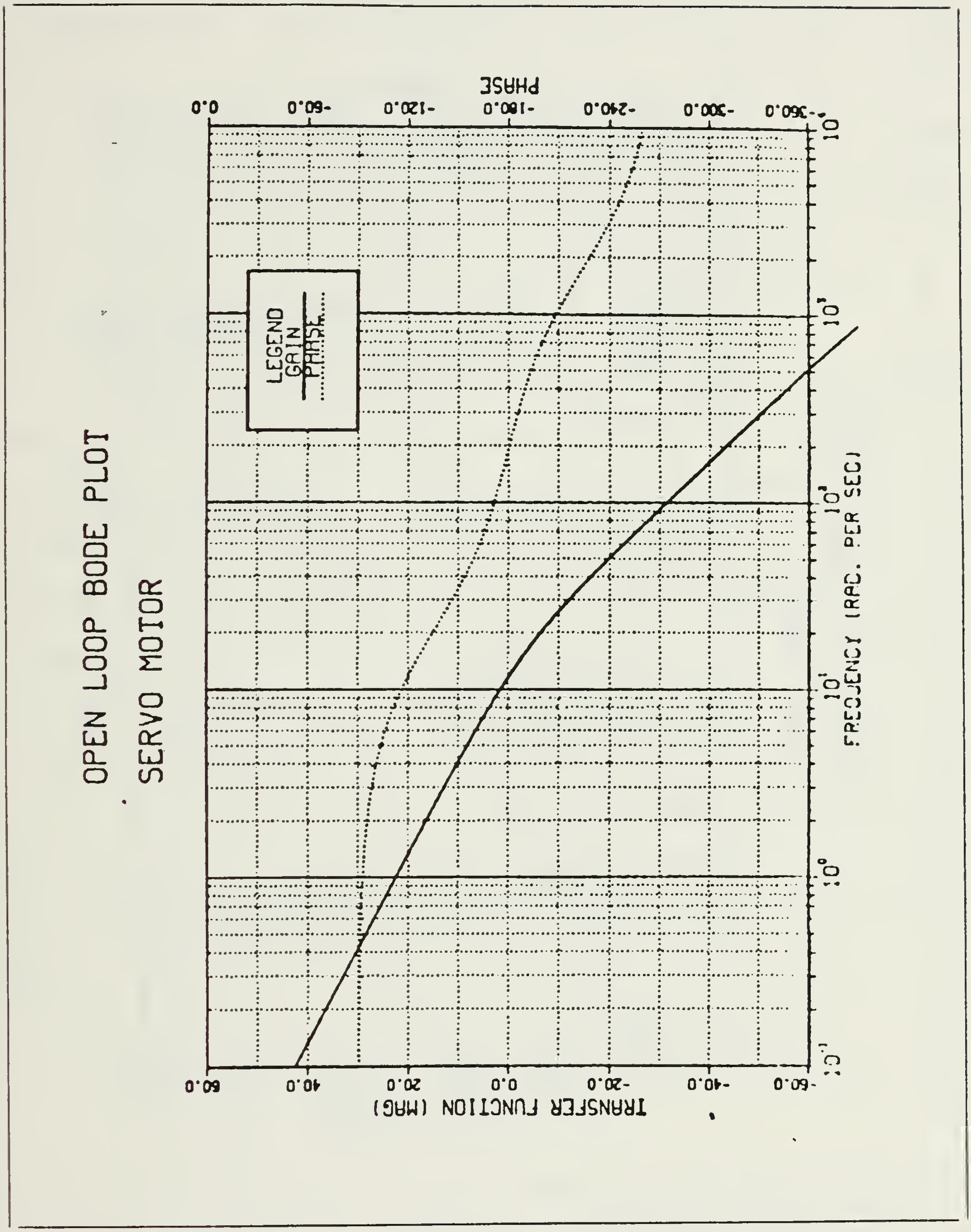

Figure 3.1 Open Loop Bode Plot of the Servo Motor Selected for Test

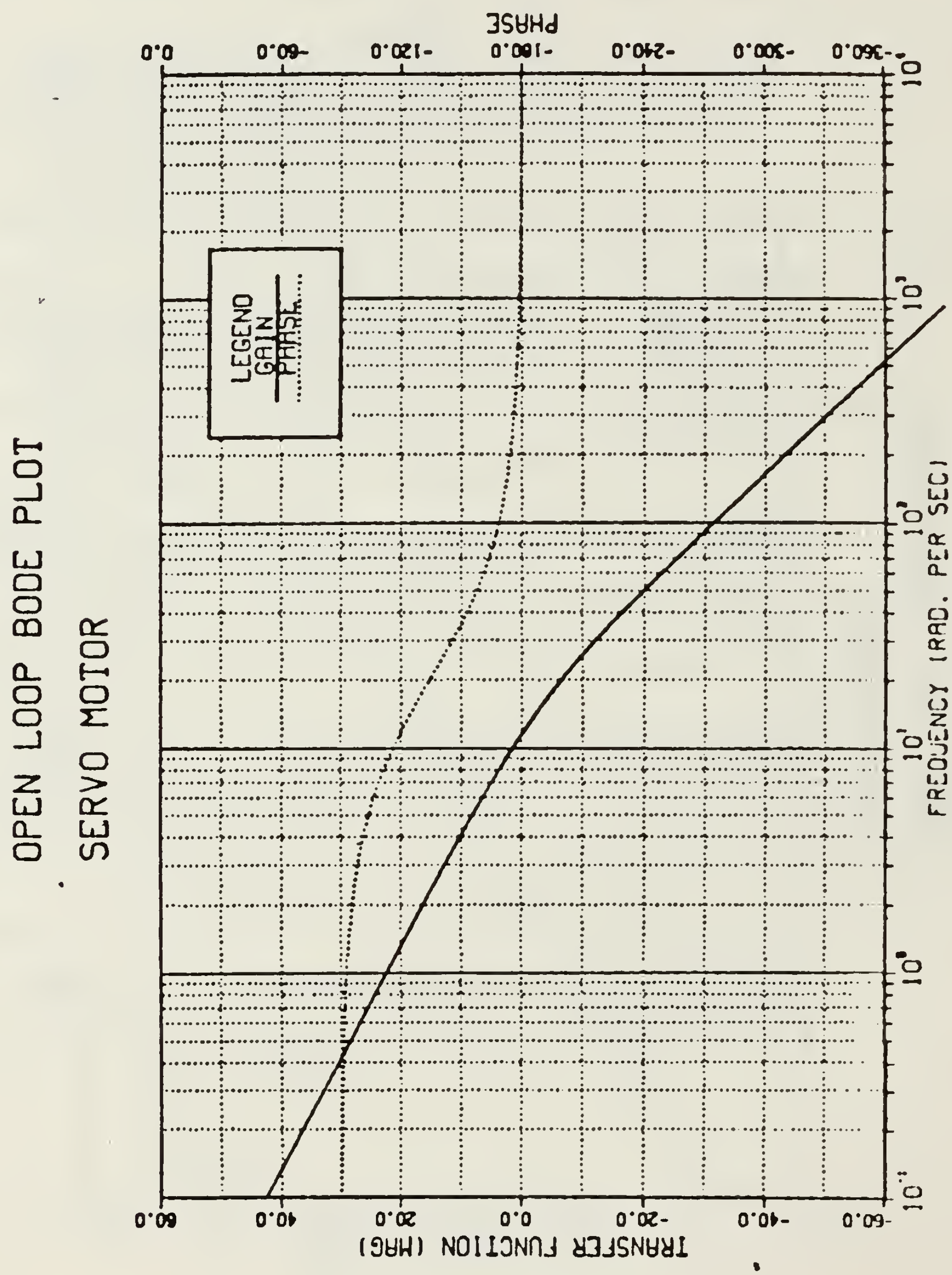

Figure 3.2 Open Loop Bode Plot with Pole at Pe Removed

magnitude response curve of Figure 3.2. This plot is shown in Figure 3.3. The -40dB/decade slope crosses the 0.0 dB axis at w= 17 radians/sec which corresponds to a Km of $\cong$300 radians/volt for an ideal motor. Thus the gain constant Km of the adaptive model was set initially to 300 radians/volt to determine the curve shape as derived in Chapter 2.

B. ALGORITHM TO UPDATE THE ADAPTIVE MODEL STATES AND GAIN PARAMETER

Figure 3.4 illustrates the modifications to the original simulation model block diagram of Figure 2.1 to make it adaptive in nature. The velocity error signal ($\dot{X}E$) is common to both the simulation model amplifier and the servo motor amplifier. The output of the servo motor (CS) is not measured directly but is read by the head at discrete sampling intervals as the servo information sectors pass beneath the head. From these observations of head position on the disk, an estimate of the servo motor gain parameter (Km) must be determined as well as the velocity of the head as it moves across the radius of the disk. Two requirements must be met in these calculations:

1. The calculations must be reasonably accurate to allow the model states to approximate the trajectory of the servo motor during the seek mode.
2. The calculations must be simple in nature to allow for the updating of model states and gain parameter in minimum time, i.e., shorter computation time delay.

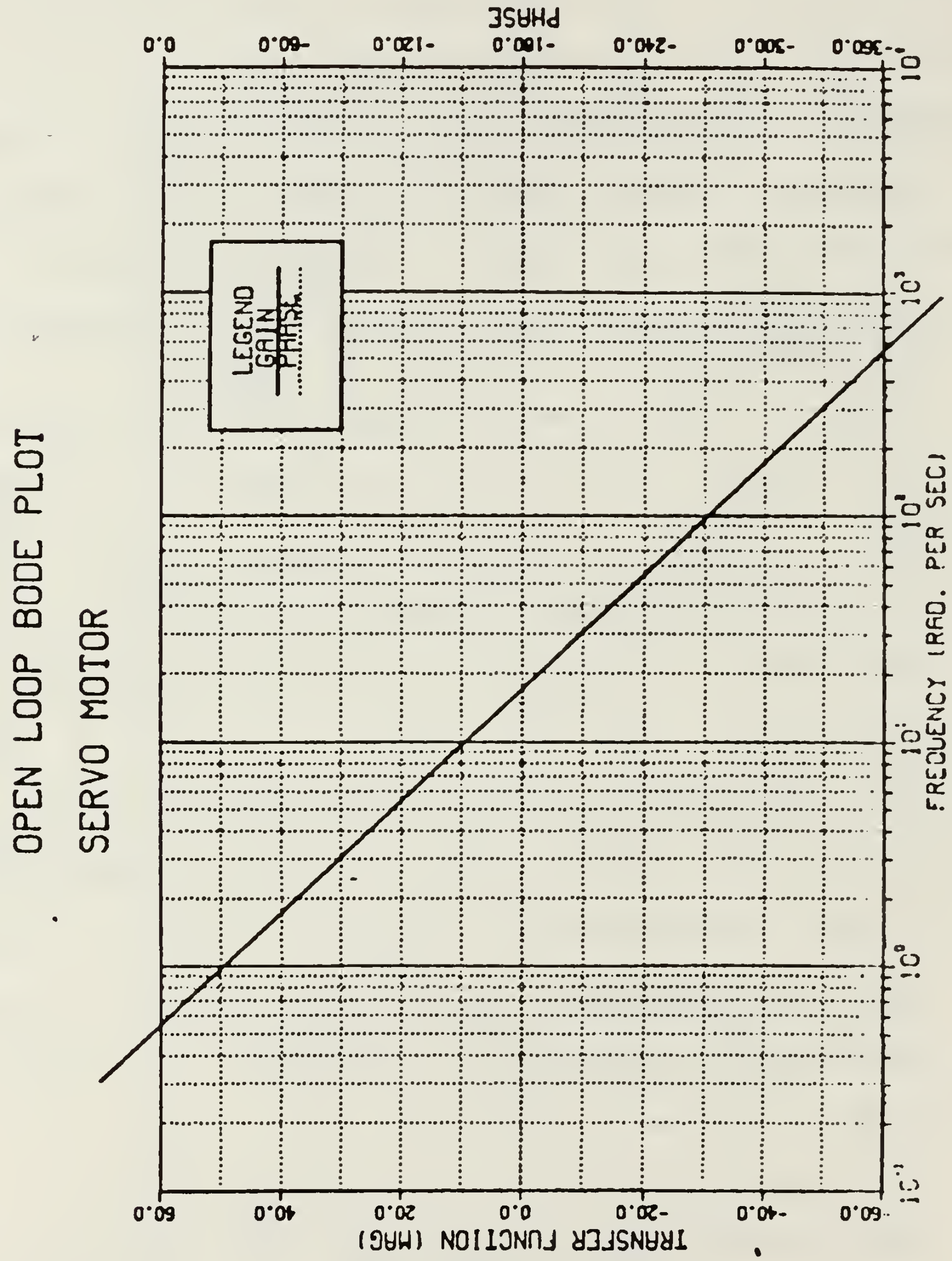

Figure 3.3 Open Loop Bode Plot for the Km/s^2 Motor

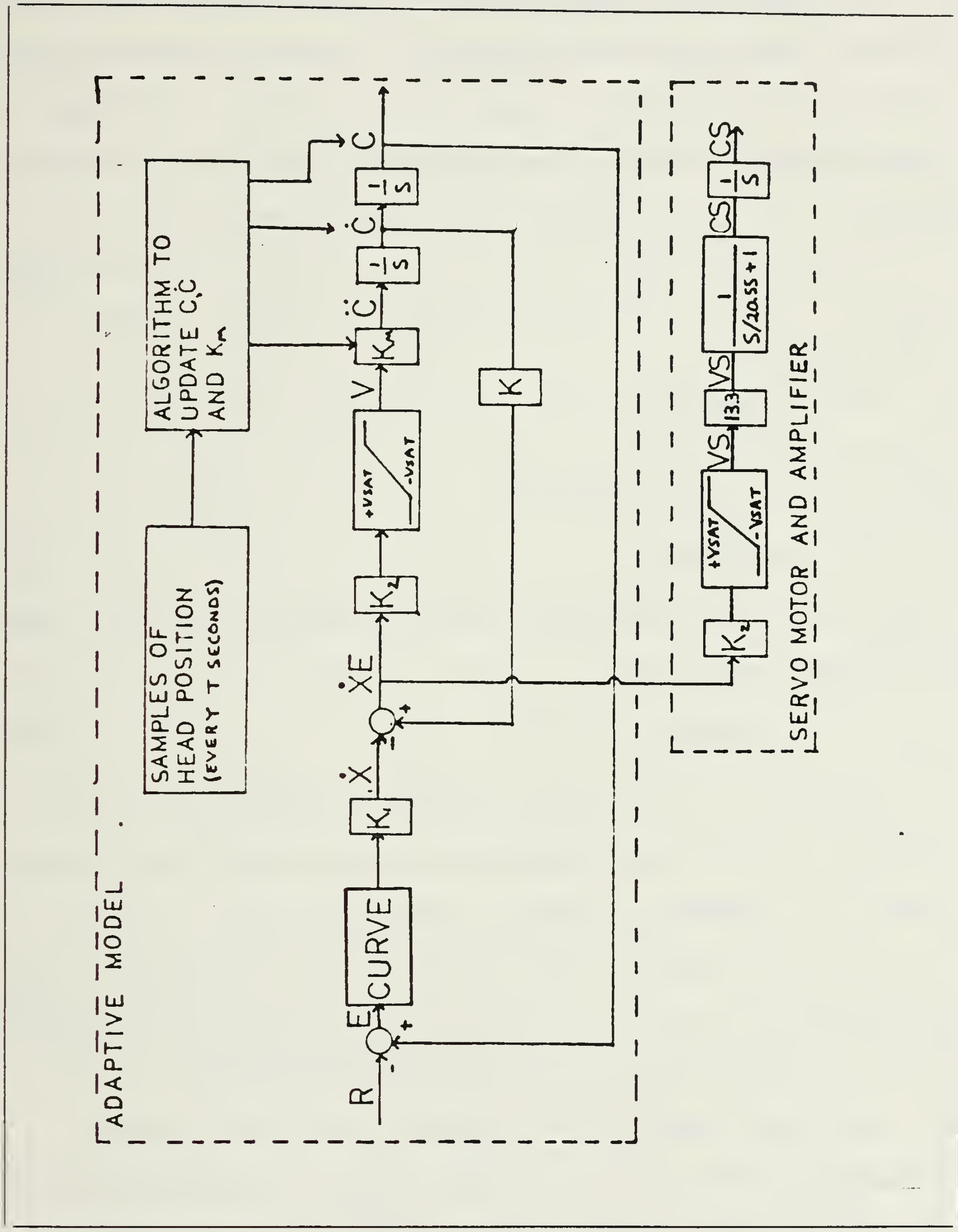

Figure 3.4 Adaptive Model Block Program

To update the adaptive model motor gain constant, the model motor requires an estimate of Km as if the non-ideal servo motor is an ideal motor, i.e., fitting the deceleration curve of the model motor to that of the servo motor. From equation 2.3 for the ideal motor

$$C = Km\ Vsat\ \frac{t^2}{2} \tag{2.3}$$

Solving for Km

$$Km = \frac{2C}{Vsat\ \ t^2} \tag{3.4}$$

For discrete time intervals

$$Km = \frac{2C}{Vsat\ (NT)^2} \tag{3.5}$$

Where T is the sampling period and N is the sample number. Letting C = Cs

$$Km = \frac{2CS}{Vsat\ (NT)^2} \tag{3.6}$$

Equation 3.6 is valid only for the full acceleration portion of the seek mode when the acceleration of the servo motor is constant. Thus, the value of Km is set for the model from samples of servo motor position during the full acceleration portion of the seek mode then remains constant for the curve following portion of the move.

Several means of estimating the head velocity were attempted such as reduced order discrete observers, Kalman filters, and direct calculation by solving for the deceleration curve of a non-ideal motor. But all these schemes failed either due to the switching from full forward

to full reverse drive to follow the curve or due to the small number of samples that are obtained during a track to track move (typically less than 4 samples). The first method that worked reasonably well is an average velocity calculation. For small sampling periods, the average velocity of the head can be estimated by:

$$\dot{C}S\ (avg) = \frac{\Delta CS}{\Delta t} = \frac{CS\ (N) - CS\ (N-1)}{T} \tag{3.7}$$

It will be shown through simulation studies that this estimate works best for long moves but close curve following is not achieved for moves of 10 tracks or less. At discrete time instants (NT) this estimate gives the average velocity over the last sampling period and not an estimate of the velocity of the head at time NT. Using equation 3.7 and another definition of average velocity

$$\dot{C}S(avg) = \frac{\dot{C}S(N-1) + \dot{C}S(N)}{2} \tag{3.8}$$

and setting equations 3.7 and 3.8 equal

$$\dot{C}S(avg) = \frac{CS(N) - CS(N-1)}{T} = \frac{\dot{C}S(N-1) + \dot{C}S(N)}{2} \tag{3.9}$$

Solving for $\dot{C}s(N)$

$$\dot{C}S(N) = \frac{2[CS(N) - CS(N-1)]}{2} - \dot{C}S(N-1) \tag{3.10}$$

This calculation provides an estimate of the head velocity as soon as the position is read from the disk and requires the storage of the last position measurement (CS(N-1)) and velocity estimate ($\dot{C}$ (N-1)). After 2 samples of position are known, the stored value of $\dot{C}S(N-1)$ can be

refined by

$$\dot{CS}(N-1) = \frac{CS(N) - CS(N-2)}{2T} \qquad (3.11)$$

This calculation allows the velocity estimate of equation 3.10 to be a very close estimate of the head velocity at the discrete sampling instant NT. The only difficulty encountered with this scheme was during a track to track move where a small number of position samples are taken. If the model begins to curve follow (switch from full acceleration to full deceleration) between samples of position, the velocity calculation of equation 3.10 is not self-correcting until two position samples can be taken after the switch. To correct this discrepency, the algorithm used detects the switch in the drive signal and stores the present value of model velocity as CS(N-1) to be used in the next calculation. By this slight modification, close curve following can be achieved even for the track to track move.

D. SIMULATION STUDIES OF THE ADAPTIVE MODEL

Appendix B list the DSL/VS simulation program used in the testing of the adaptive model and is a modification of the program of appendix A. A sample region was added to simulate the reading of head position information at discrete time intervals CS(N). The DELS parameter of the program allows the simulation to enter the sample region every T = 0.25 milliseconds where a sample of head position,

CS , is taken. This corresponds to approximately 64 sectors of servo information being interspersed in the data tracks of a disk file rotating at 3600 rpm as discussed in Chapter 1.

From the sample of head position the algorithm, also contained in the sample region, computes the estimate of $\dot{CS}$ and the gain parameter Km. Model states C, $\dot{C}$ are reset with the values of CS and $\dot{CS}$ and the gain parameter Km is updated during the full acceleration portion of the move. This program uses equation 3.7 as the calculation for $\dot{CS}$.

Fixed step integration was used in the simulation as this would be the easiest numerical integration routine to implement in a microprocessor. The integration step size, parameter DELT, was chosen to be 1/5 the sampling period to allow 5 integration steps between update of the adaptive model. The model velocity error signal, $\dot{XE}$, is computed at each integration interval and is sent to the servo motor amplifier.

Figures 3.5 and 3.6 are the phase plane plot and step response curves for a step command of 100 tracks. In the phase plane plot, the model velocity $\dot{C}$ and the head velocity $\dot{CS}$ are plotted versus model position C. Although $\dot{CS}$ can not be observed in the actual system, it is available in the simulation and is used to check the validity of the calculations in the algorithm. Figure 3.5 shows that the adaptive model and servo motor have good curve following

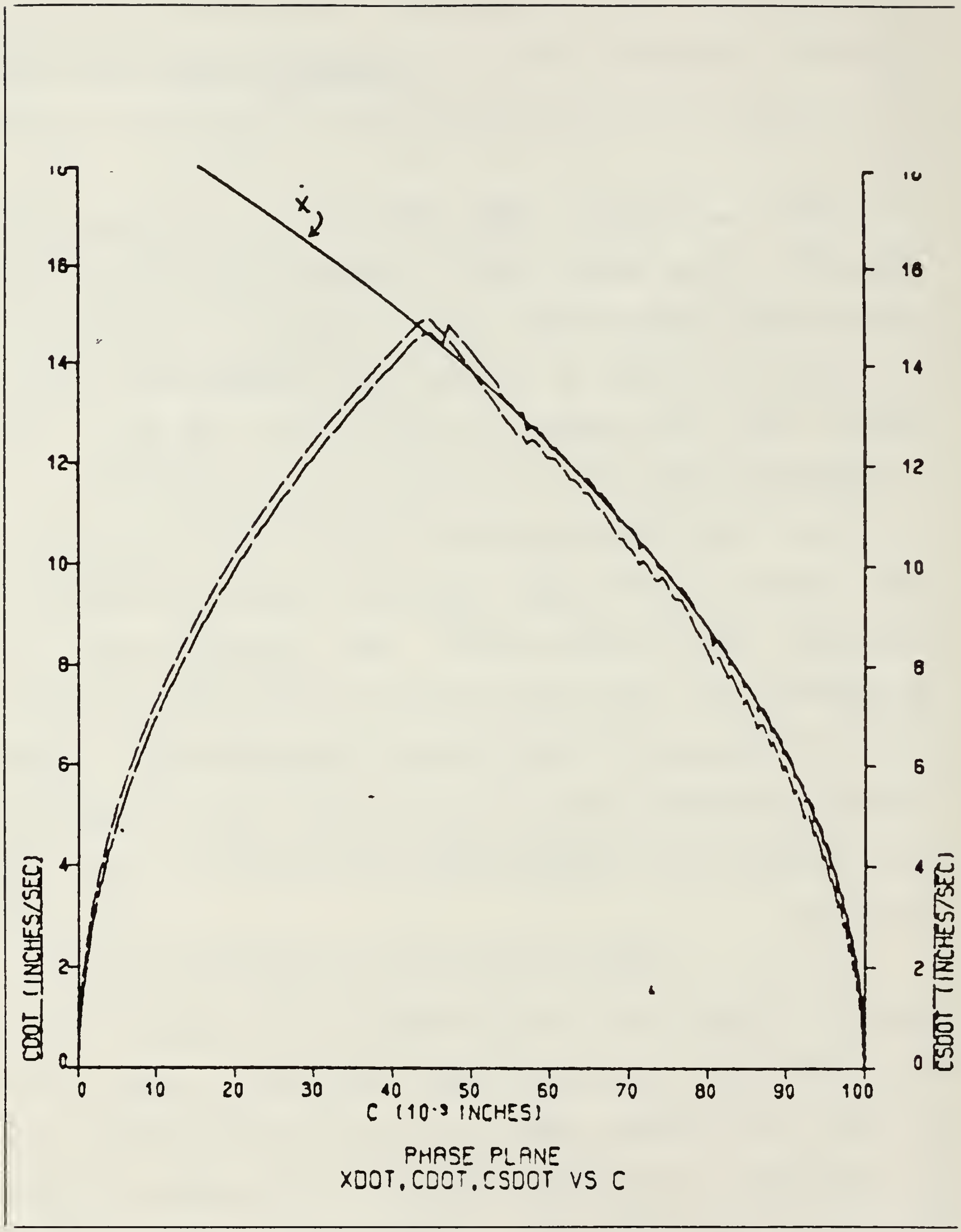

Figure 3.5 Phase Trajectory of Adaptive Model
Using Average Velocity Updates
(100 Track Move)

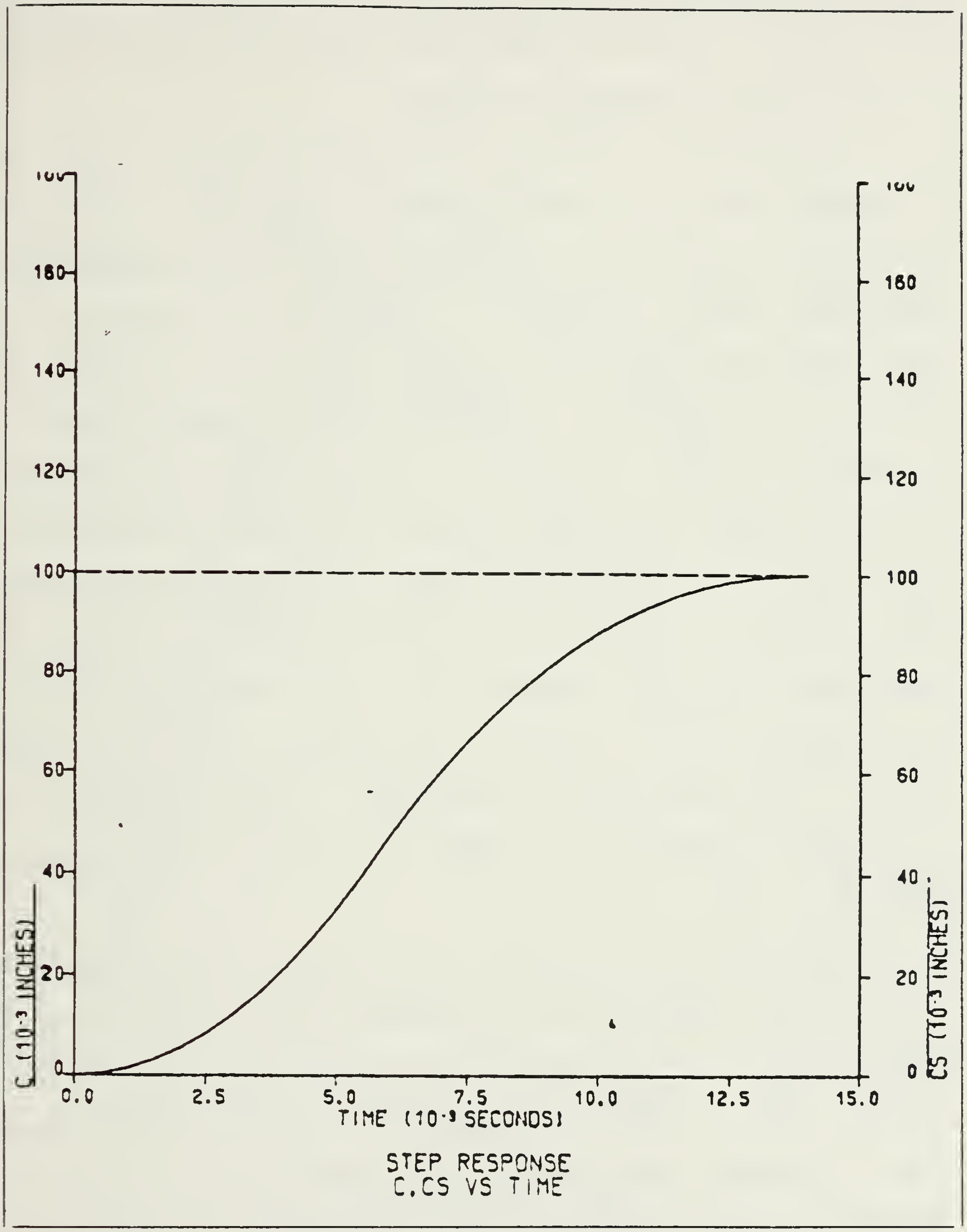

Figure 3.6 Step Response of Adaptive Model
Using Average Velocity Updates
(100 Track Move)

characteristics for the long move. The step response curve of Figure 3.6 shows the model and servo motor track together to the desired commanded position.

Figures 3.7 and 3.8 are the phase plane plot and step response curves for a step command of one track. The phase plane plot illustrates the problem encountered when updating the model velocity state with the $\dot{C}S(avg)$ calculation. When the desired position is reached, the velocity of the head does not reach zero but is still moving at approximately 0.8 inches/sec. This velocity will be of concern when switching into the track follow mode because it tends to increase both the overshoot and settling time of the compensated system. The updating of the adaptive model can be seen in the step response curve of Figure 3.8. At multiples of 0.25 milliseconds, the curves come together then drift apart until the next update. This drift is attributed to the inaccurate updating of head velocity at the sampling instant.

By using equation 3.10 as the estimate of head velocity along with equation 3.11 to correct the previous velocity calculation, close curve following can be achieved for any length move. Appendix B also lists the simulation program that reflects the change in the algorithm to improve the velocity calculation and Figure 3.9 shows the phase plane plot for the 100 track move utilizing this new algorithm. The trajectory of both the model and servo are practically

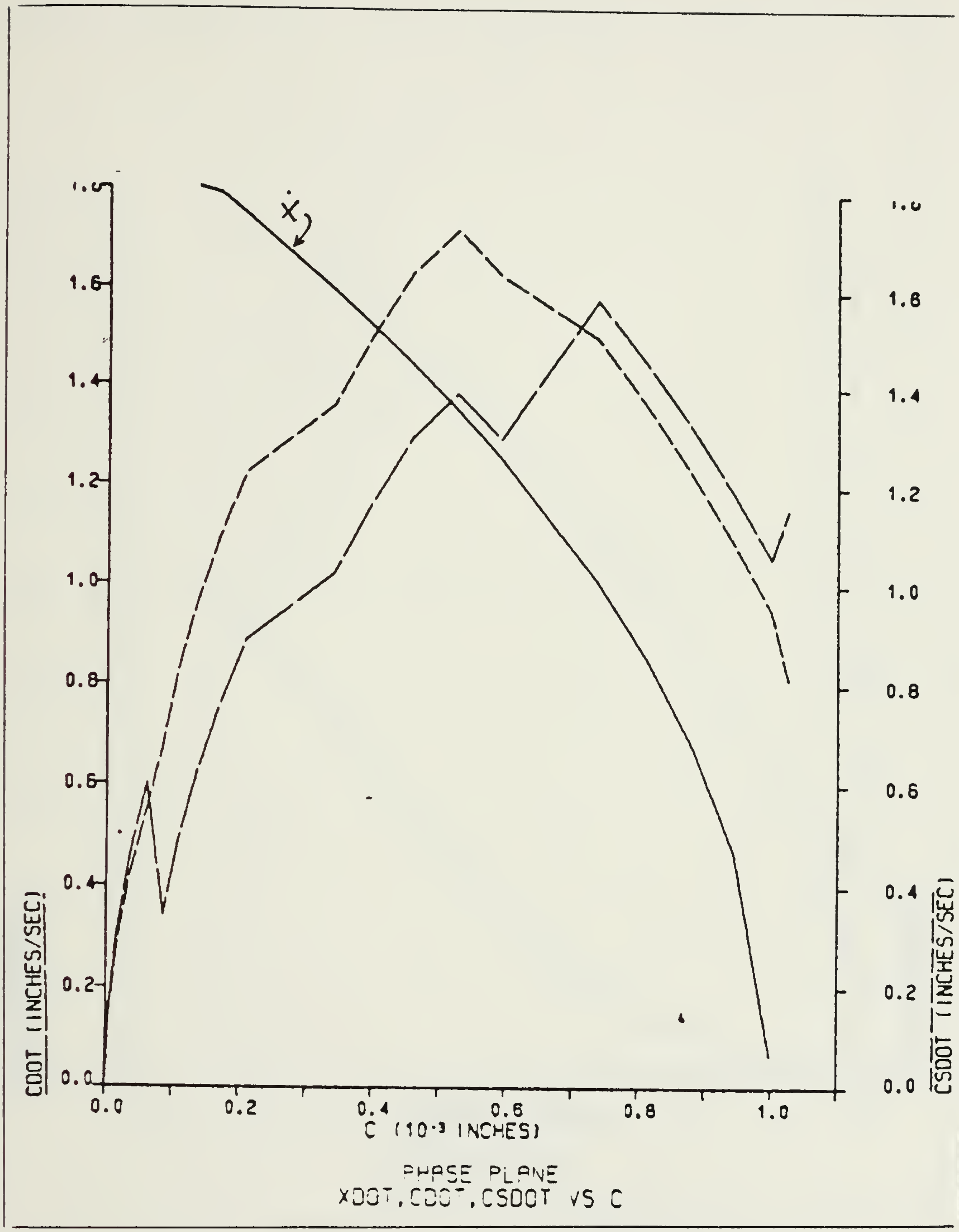

Figure 3.7 Phase Trajectory of Adaptive Model
Using Average Velocity Updates
(1 Track Move)

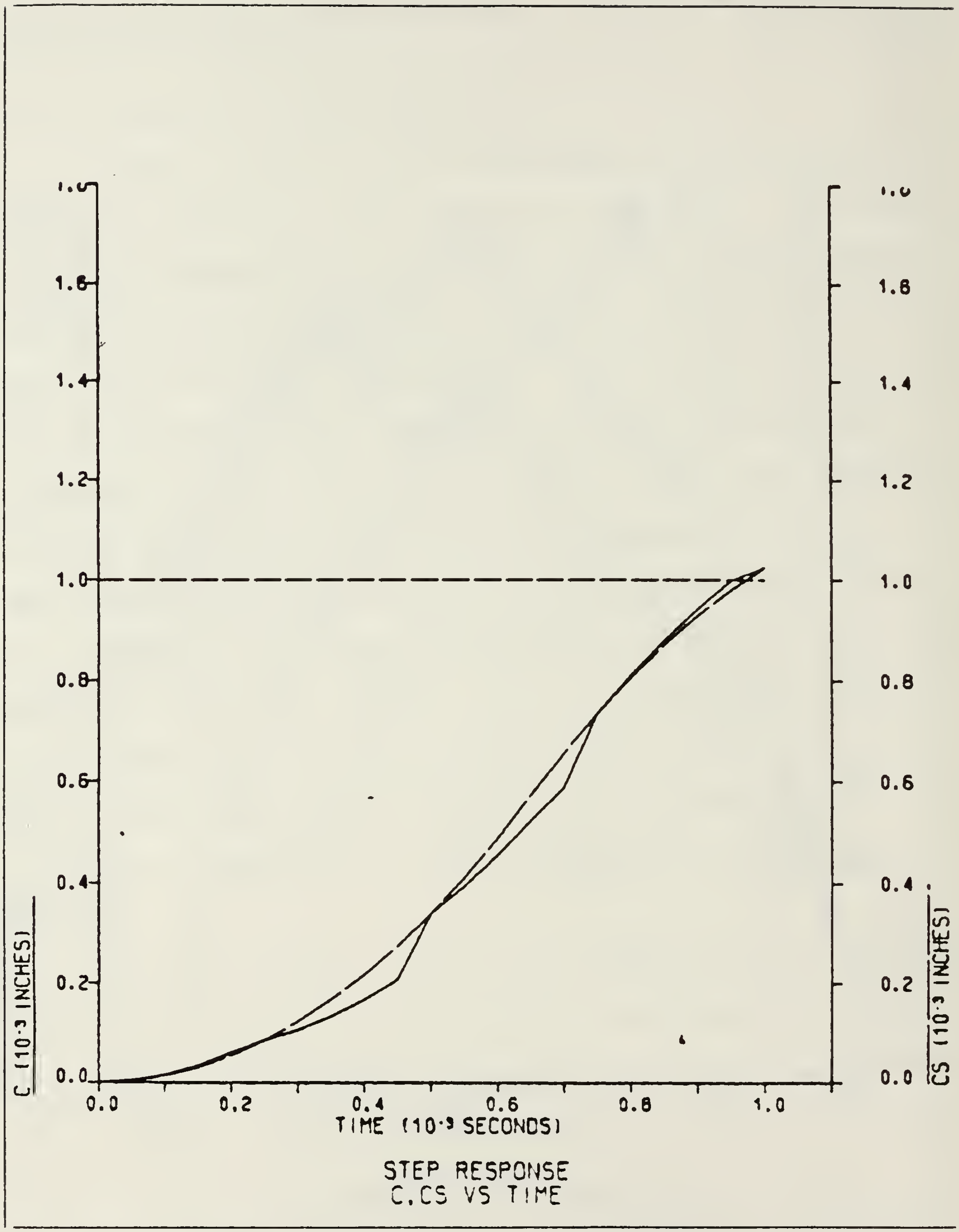

Figure 3.8 Step Response of Adaptive Model
Using Average Velocity Updates
(1 Track Move)

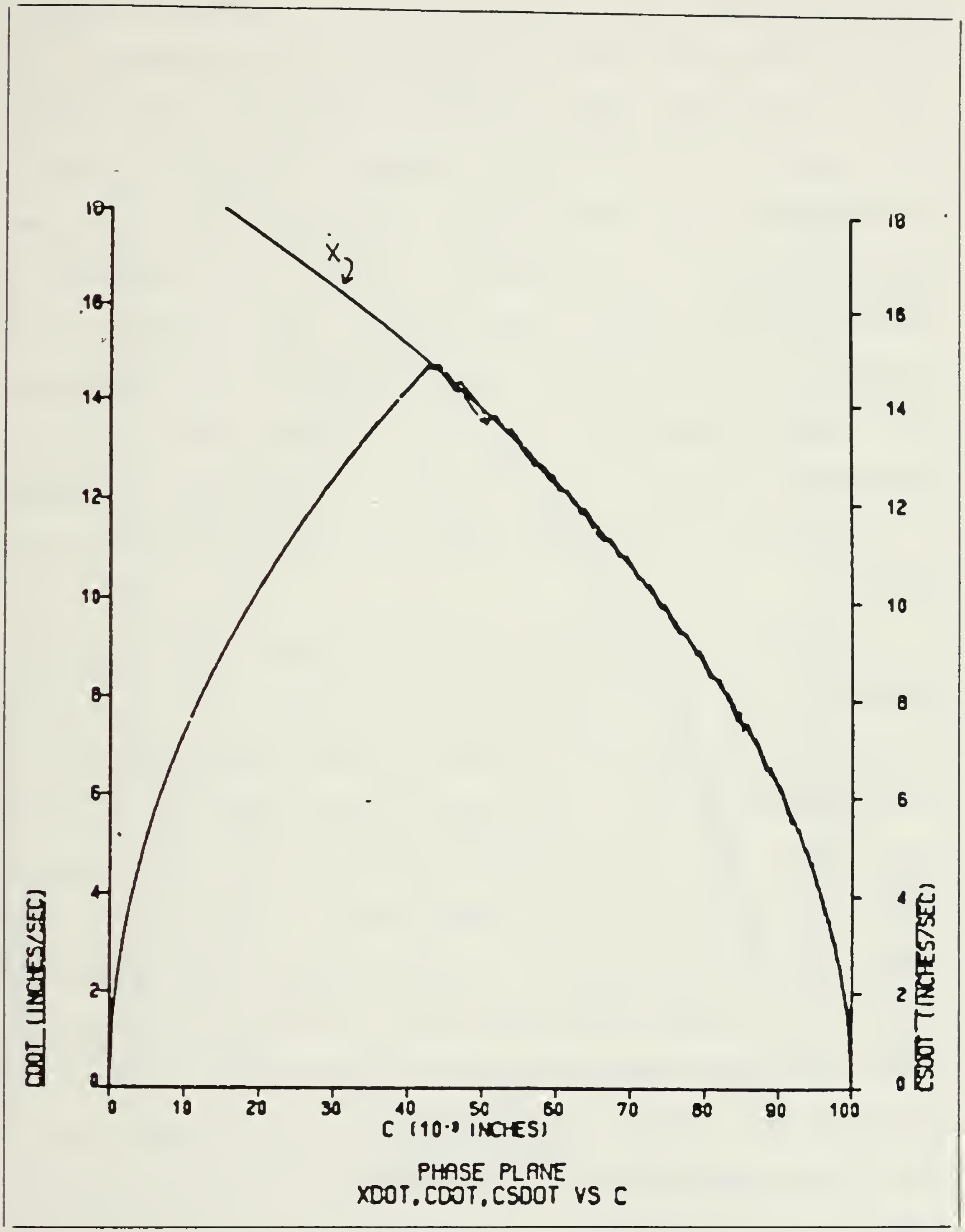

Figure 3.9 Phase Trajectory of Adaptive Model Using Velocity Calculation of Equation 3.10 (100 Track Move)

identical and close curve following is achieved. Figure 3.10 is the phase plane plot for the one track move. It is easy to see that after the first sample of head position, the model and servo track together during both the full acceleration and deceleration portions of the seek mode. Head velocity is noticeably reduced to less than 0.2 inches/second when the head is over the desired position. Figure 3.11 illustrates the improvement in step response of the adaptive model for the track to track move. Based on the results shown in these figures, the program of Appendix B was chosen as the base program for further studies in this thesis.

Included in Appendix B is the DSL/VS simulation data printout for the 100 track move of Figures 3.10 and 3.11. At the first sampling instant and every multiple of 0.25 milliseconds later, the state points C and $\dot{C}$ of the model are reset to the observed position of the head and the calculated head velocity respectively. It is evident that the velocity calculation is quite accurate throughout the move. It is also seen that after the initial adjustment of Km, this value does not change appreciably over the acceleration portion of the move and then is held constant during the deceleration part of the move. This resetting of model states and adjustment of Km at each sampling instant allows the model trajectory to closely match that of the servo between samples. This ability to track the servo is

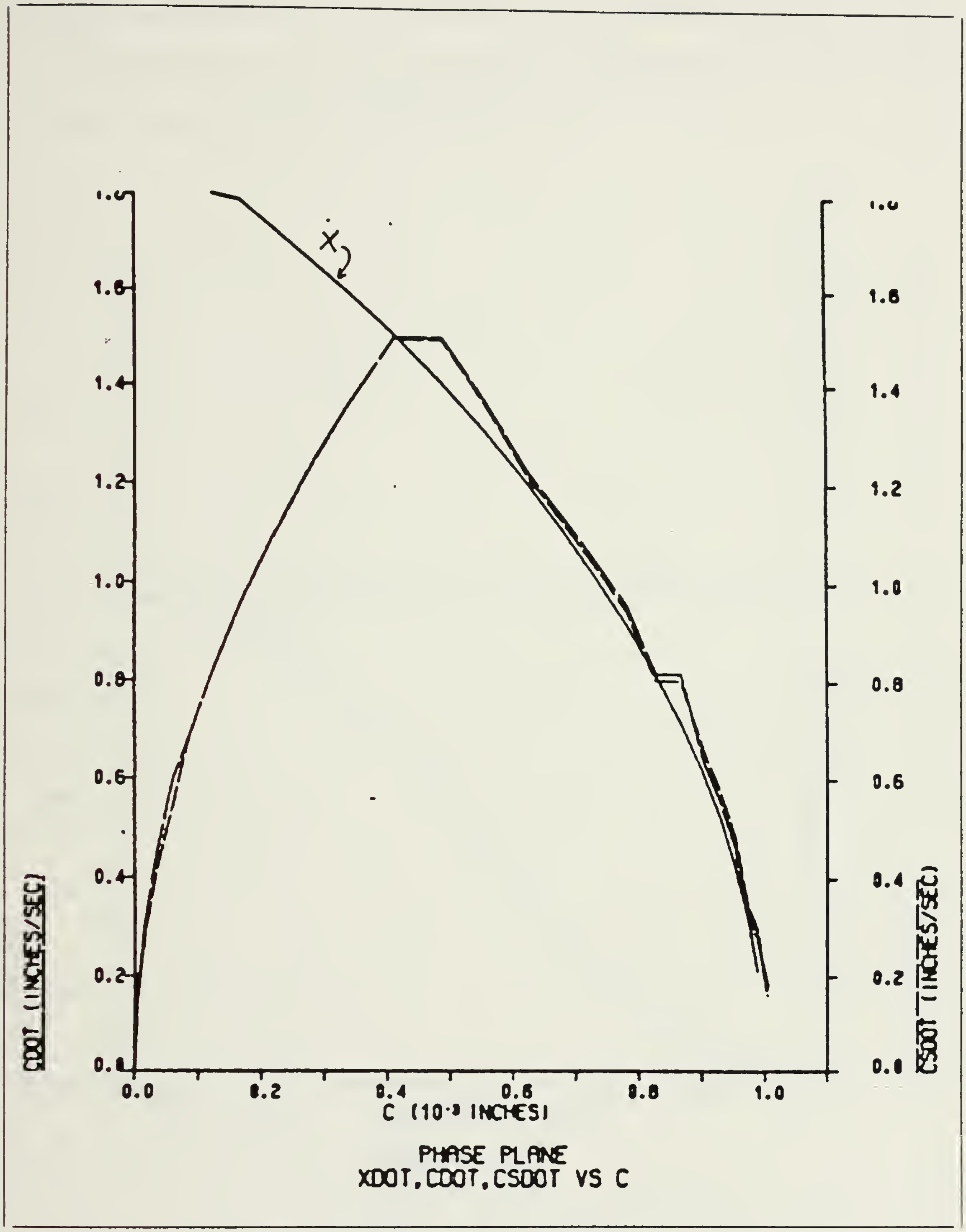

Figure 3.10 Phase Trajectory of Adaptive Model
Using Velocity Calculation of Equation 3.10
(1 Track Move)

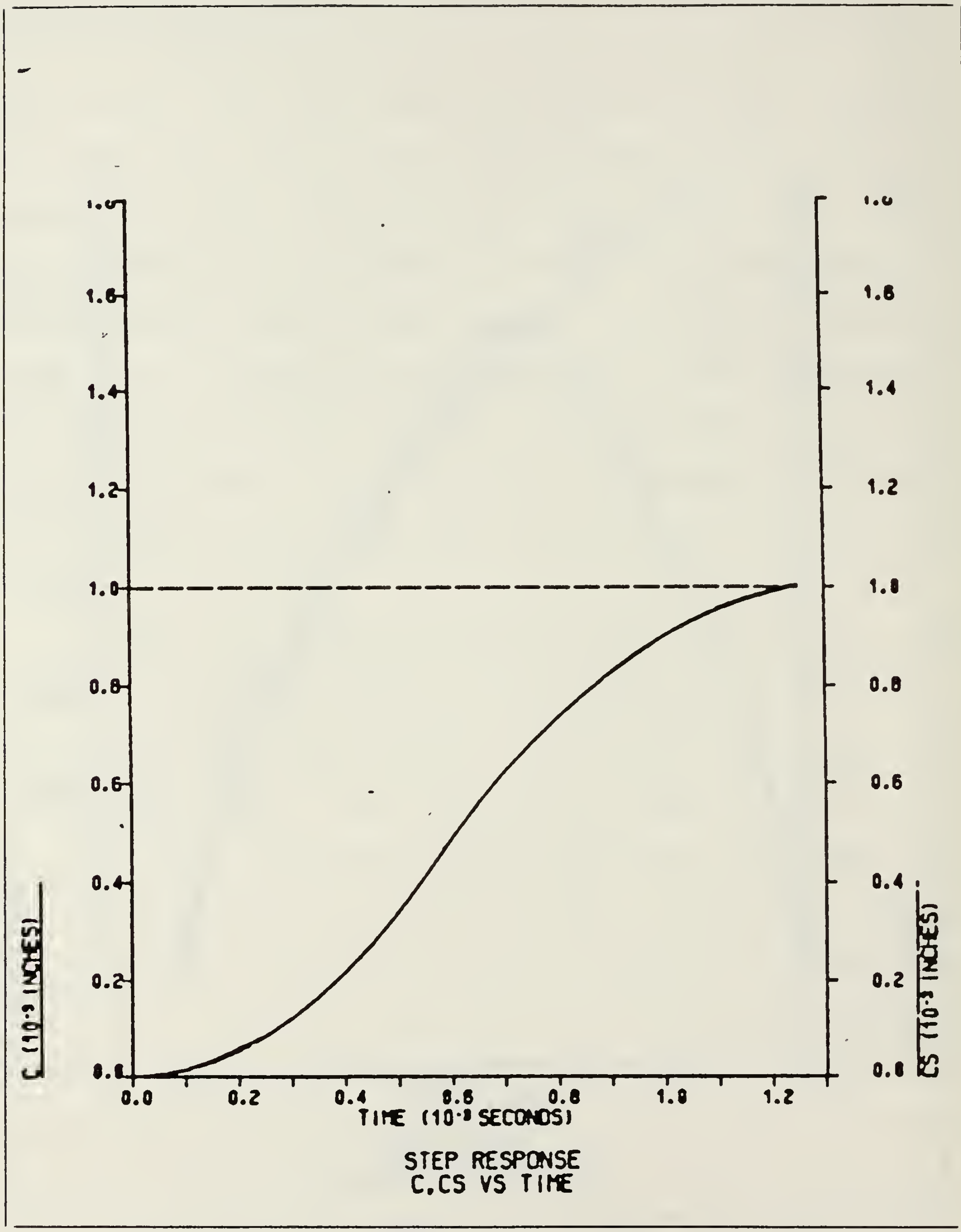

Figure 3.11 Step Response of Adaptive Model
Using Velocity Calculation of Equation 3.10
(1 Track Move)

the primary factor in the close curve following ability of the model and servo combination and reduces the velocity of the head when the desired position is reached.

IV. ADDITIONAL TESTING OF THE ADAPTIVE MODEL DURING THE SEEK MODE

A. EFFECT OF CHANGING AMPLIFIER GAIN PARAMETER K2

As discussed in Chapter 2, the linear gain of the amplifier, K2, is chosen to saturate the amplifier for moves of one track during the full acceleration portion of the seek mode and to allow for close curve following to the desired position. Figures 4.1, 4.2 and 4.3 are the one track move phase plane plots for K2 = 1000, 100 and 10 respectively. For the first two cases, the value of K2 saturates the amplifier and both full forward drive and close curve following are achieved. But for K2 = 10, the amplifier saturates initially but drops out of saturation at the peak of the trajectory and both the model and servo motor fail to curve follow. Thus the amplifier gain must be chosen to be 100 or greater to obtain the desired phase trajectory for the one track move.

B. EFFECT OF CHANGING AMPLIFIER SATURATION LIMIT VSAT

The saturation limit, Vsat, of the amplifier is determined by the available power supply voltage and affects the phase trajectory in two ways. First it determines the magnitude of the full forward and reverse drive applied to the model and servo motor and thus affects the time required for the desired move (seek time). Secondly it scales the curve to be followed. Recalling equation 2.10, the

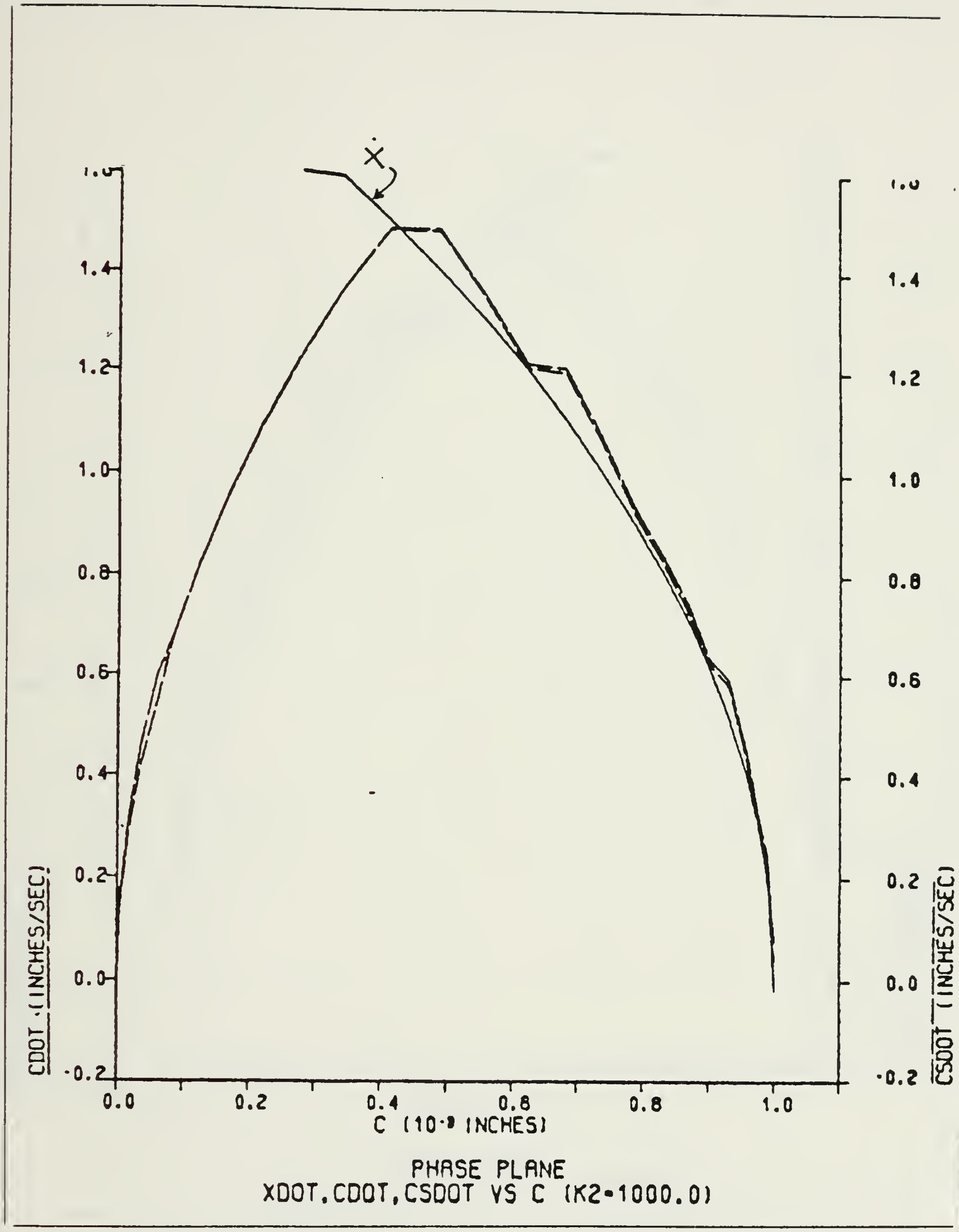

Figure 4.1 Phase Plane Tracjectory, K2 = 1000

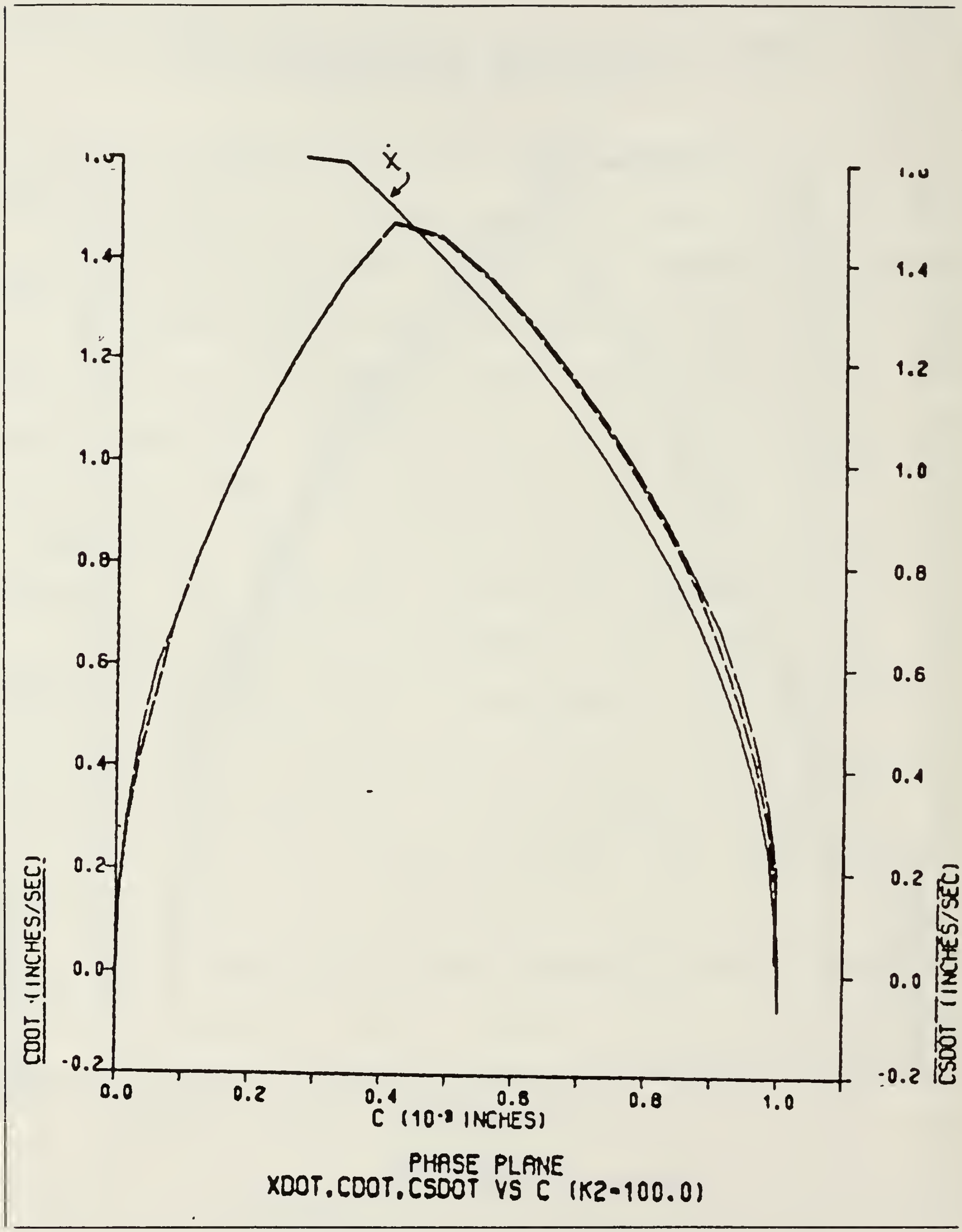

Figure 4.2 Phase Plane Trajectory, K2 = 100

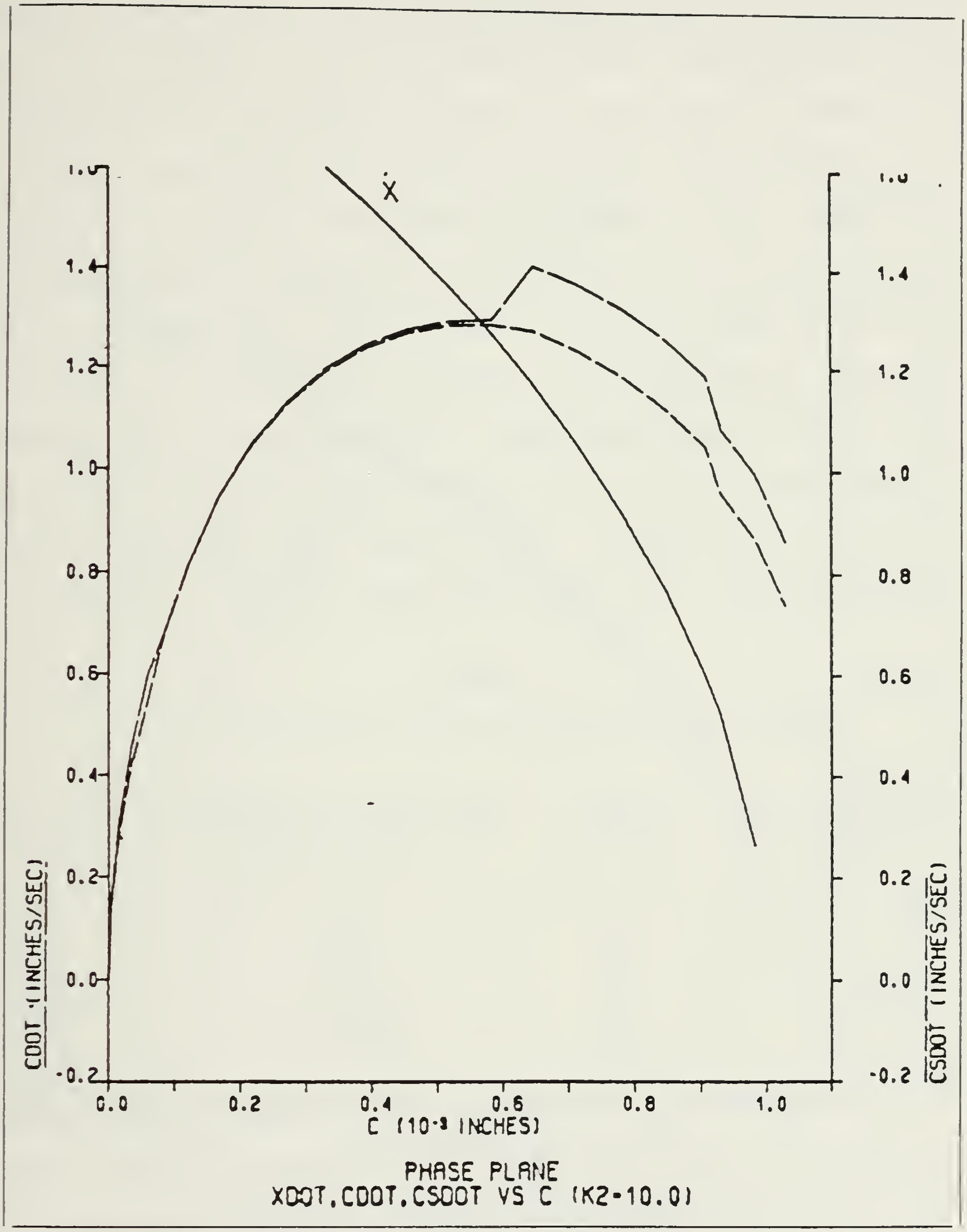

Figure 4.3 Phase Plane Trajectory, K2 = 10

parameter A multiplies the square root of the error signal and determines the commanded velocity curve, $\dot{X}$, to be followed. The curve is scaled up when the saturation point is raised and scaled down when the saturation point is lowered. This effect is illustrated in the one track move phase plane plots of Figures 4.4 and 4.5 for Vsat equal to 5.0 volts and 15.0 volts respectively. By comparing these figures with Figure 3.10 (Vsat = 10.0 volts) the scaling of the X curve can be observed. The maximum velocity achieved by the servo motor is also noticed to increase as the value of Vsat is increased. These effects are tabulated in Table 1 for both the 1 and 100 track moves with K2 = 10,000.

TABLE 1

Effect of Varying Amplifier Saturation Voltage

Length of Move (tracks)	Vsat (volts)	Max Velocity Achieved (inches/sec)	Seek Time (msec)
1	5	1.06	1.75
1	10	1.50	1.25
1	15	1.77	1.00
100	5	10.35	18.90
100	10	14.63	13.50
100	15	18.15	10.90

As expected, the seek time is reduced as the maximum velocity of the servo motor is increased by raising Vsat. Thus the seek time can be adjusted by selection of the saturation level of the amplifier as determined by the available power supply voltage. But the drive on the motor

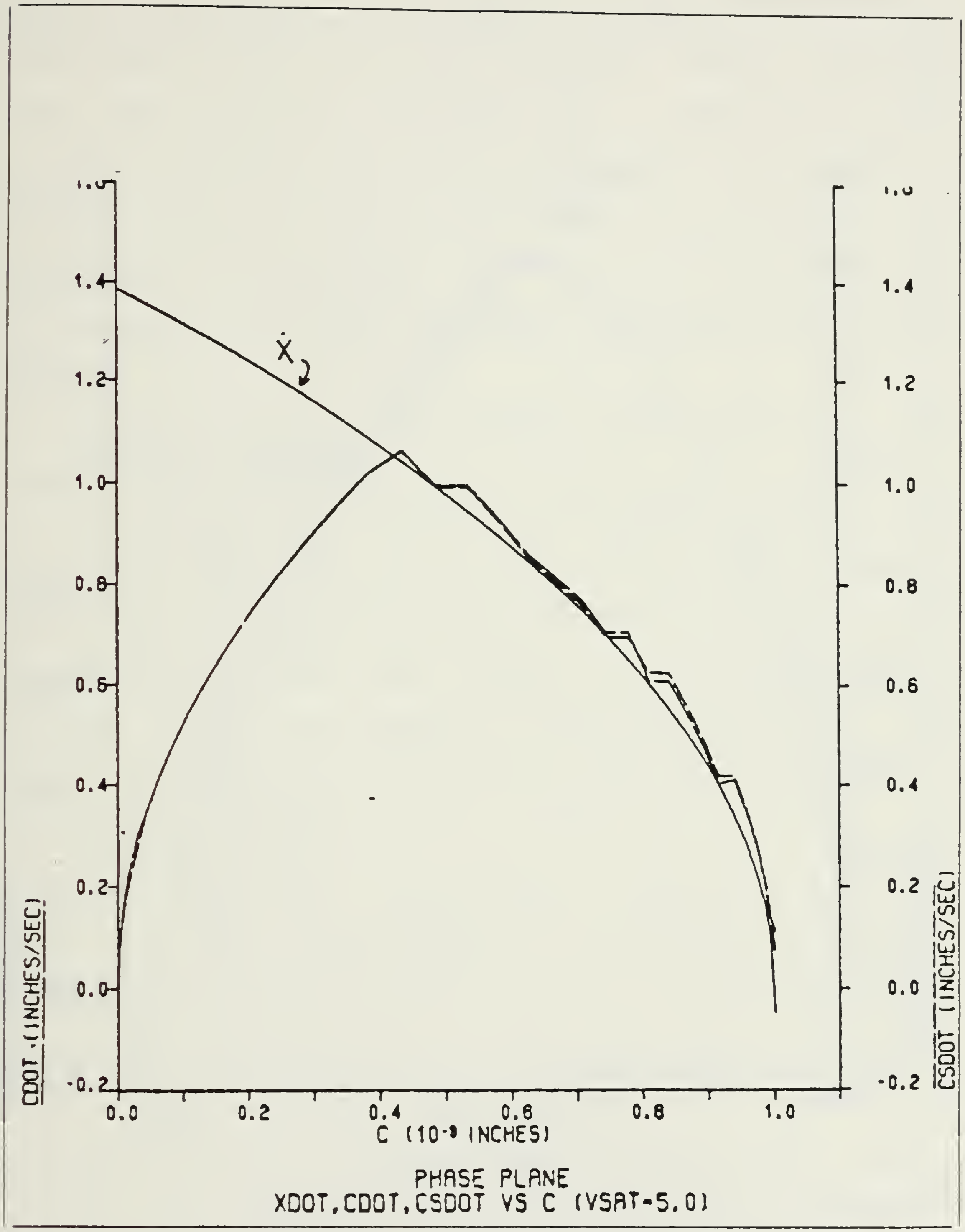

Figure 4.4 Phase Plane Trajectory, Vsat = 5.0

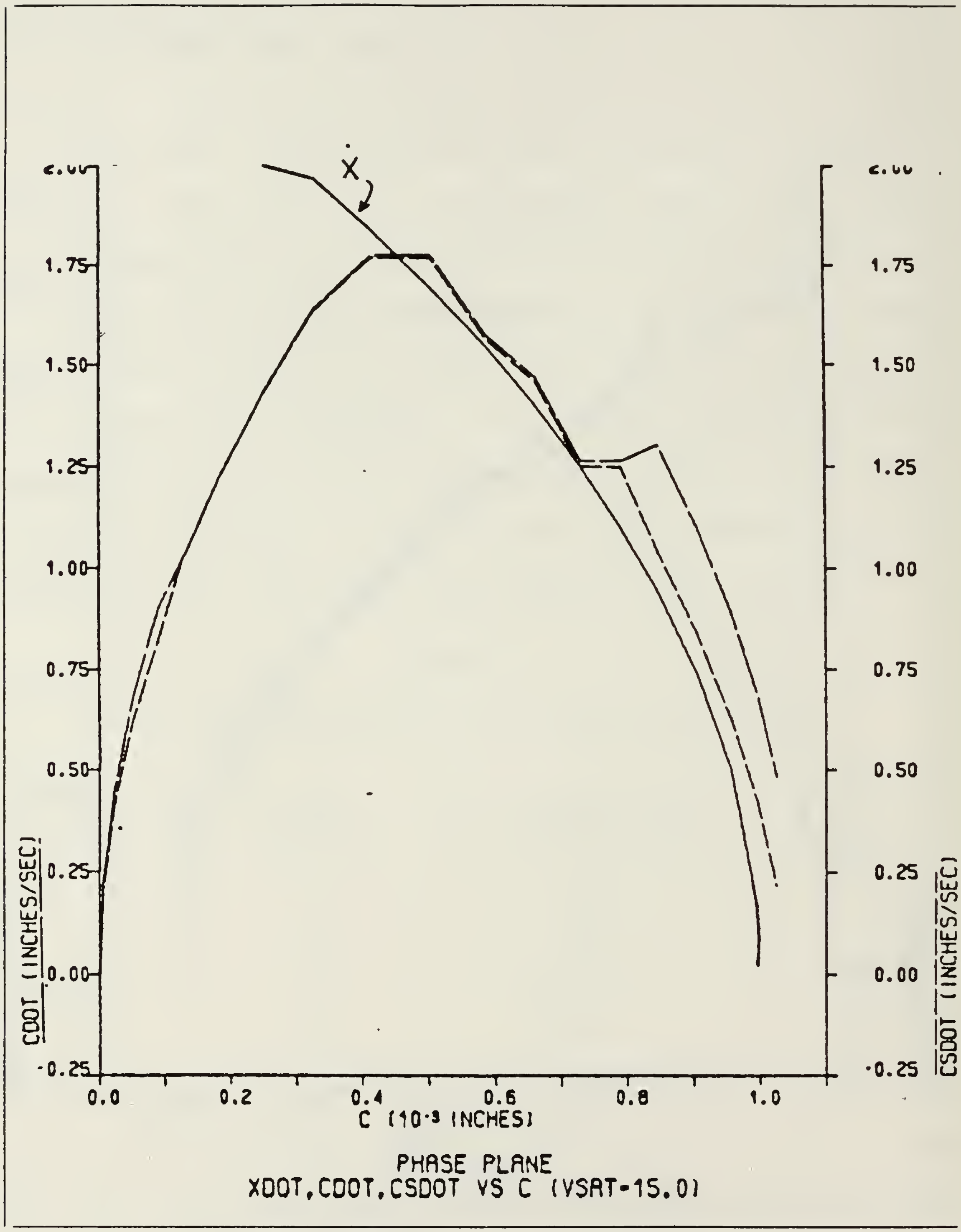

Figure 4.5 Phase Plane Trajectory, Vsat = 15.0

cannot be adjusted too high as curve following is jeopardized (Figure 4.5) when the model cannot accurately track the actions of the servo motor. The loss of curve following results when only one update of the model is obtained during the curve following portion of the move. As the drive on the model and servo motor is increased, the shorter seek time reduces the number of samples taken during the move.

C. EFFECT OF ADDING THE ELECTRICAL POLE TO THE SERVO MOTOR TRANSFER FUNCTION

In Chapter 3, the decision was made to neglect the electrical pole, Pe, from the servo motor transfer function (equation 3.2). When the electrical pole is added back into the transfer function (and likewise to the DSL/VS simulation program) significant problems arise as shown in the phase plane plot and step response curves of Figures 4.6 and 4.7 for the one track move.

Even though full forward drive is applied to the servo motor, it does not accelerate at a constant rate like the model and as a result the velocity of the servo motor builds up slowly at first. At the first model update (t=.25 msec) the states are reset in the model but because the calculation of Km is based on constant acceleration between samples, the value of Km is low and the model accelerates more slowly than the servo motor over the next time interval. This can be seen in the step response curve of

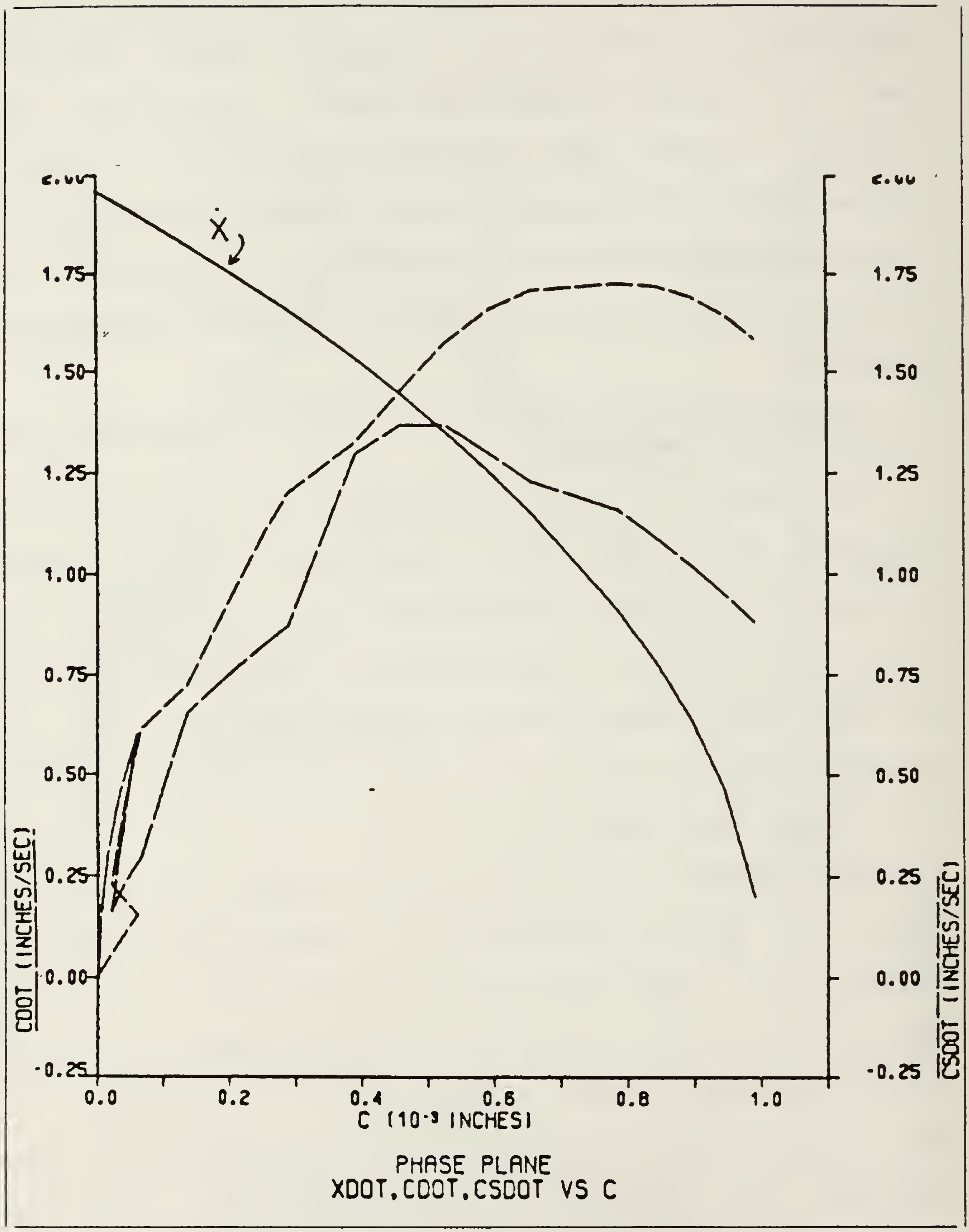

Figure 4.6 Phase Plane Trajectory of 3rd Order Servo Motor

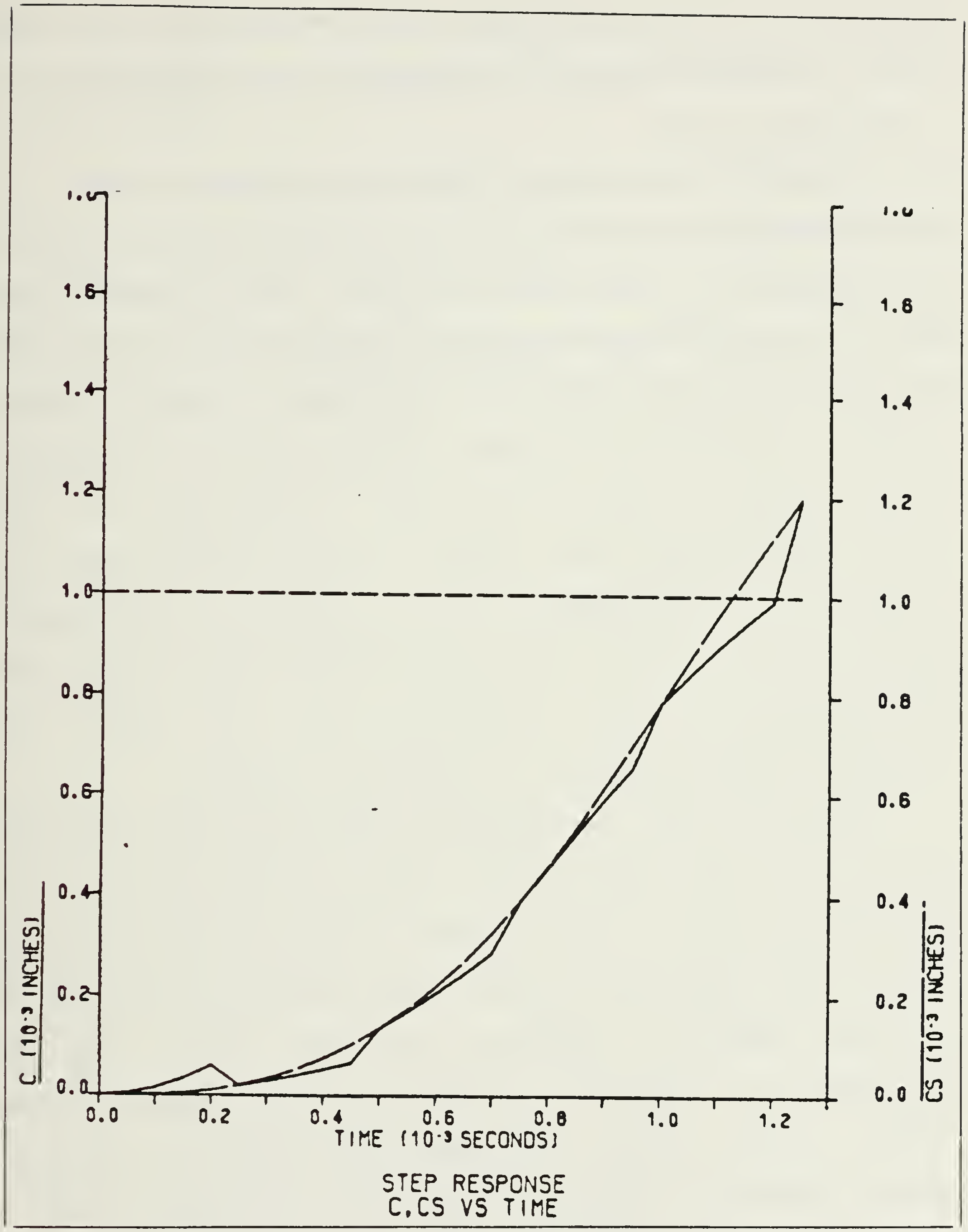

Figure 4.7 Step Response of 3rd Order Servo Motor

Figure 4.6. When switching to full reverse drive, the servo motor cannot decelerate as quickly as the model and curve following is lost.

Because the electrical pole is determined by the ratio of the armature resistance to the armature inductance, it is the inductance of the armature circuit that delays the build up of armature current and hence full torque is not applied to the motor until the electrical time constant of the armature circuit is overcome. For this motor the electrical time constant (1/Pe) is 0.283 msec. As a rule of thumb, four time constants must elapse in an R-L circuit before the current can be considered to be at a maximum. This explains why the model and the servo motor do not track together when the electrical pole is added back into the servo motor transfer function making it a 3rd order system.

Thus, the motor selected for testing in this thesis may not be the best choice to use a when voltage source is the desired drive for the servo motor. Curve following for the short move cannot be achieved with this servo motor for the desired speed of response and high track densities discussed. To meet these design parameters with a voltage source drive system, the servo motor armature inductance must be reduced. Current technological advances in the DC servo motor industry indicate that reduction of the inductance of the armature circuit to negligible amounts is possible. Therefore, it is assumed that this requirement

can be met and this thesis will continue to use the second order model of the servo motor in the development of a track follow scheme.

Before moving to the track follow mode, an alternative type of servo motor drive will be presented to show how the 3rd order servo motor can be made to respond with characteristics similar to that of the ideal motor of the model. Such a system is called the current source drive system.

V. THE CURRENT SOURCE DRIVE SYSTEM

A. INTRODUCTION

Up to this point the drive on servo motor has been a voltage source drive, i.e., the amplifier supplies a voltage to the armature circuit of the DC servo motor. This has been shown not to be a good choice for the servo motor selected for test (3rd order system). To achieve constant acceleration, the current in the armature circuit must build up quickly to a constant value and apply constant torque to the load in a order for the model to control the servo motor during the seek mode. One means of accomplishing this is to use a current source drive system.

B. DESIGN OF THE CURRENT SOURCE DRIVE SYSTEM

The equations that describe a permanent magnet DC servo motor are considered first [Ref. 2]. For the motor initially at rest so that all initial conditions are zero the equations are

$$T = Kt \cdot I = J\ddot{\theta} + f\dot{\theta} \qquad (5.1)$$

$$V = IR + L\dot{I} + Kv\dot{\theta} \qquad (5.2)$$

where

V = applied d-c voltage

I = armature current

R = armature resistance

L = armature inductance

Θ = angular position of the motor shaft

T = motor output torque

Kt = motor torque constant

Kv = motor emf constant

J = inertia of motor and load

f = viscous friction of motor and load

By taking the Laplace transforms of the above equations, the transfer function from V as input to Θ as an output is

$$\frac{\Theta(s)}{V(s)} = \frac{Kt}{S[(R+sL)(Js+f) + Kv\ Kt]} \qquad (5.3)$$

The transfer function from V as an input to I as an output is

$$\frac{I(s)}{V(s)} = \frac{Js + f}{[(R+sL)(Js+f) + Kv\ Kt]} \qquad (5.4)$$

If the viscous friction of the motor and load is small it may be neglected and the above equations become

$$\frac{\Theta(s)}{V(s)} = \frac{Kt}{s[(R+sL)(Js) + Kv\ KT]} \qquad (5.5)$$

$$\frac{I(s)}{V(s)} = \frac{Js}{[(R+sL)(Js) + Kv\ Kt]} \qquad (5.6)$$

It can be shown [Ref.2] that by factoring the denominator and manipulation of equation 5.5, the equivalent transfer function of the servo motor is

$$\frac{\Theta(s)}{V(s)} = \frac{1/Kv}{s(s\frac{L}{R} + 1)(s\frac{JR}{Kv\ Kt} +1)} \qquad (5.7)$$

Consider now the current source drive system of Figure 5.1. When a voltage representing the current flowing in the armature circuit is fed back into the input circuit of a

voltage amplifier, the amplifier can be converted to a current source. The armature current feedback is provided by the voltage drop across a small resistor (RSENSE) placed in the armature circuit. It is amplified to be proportional to the armature current.

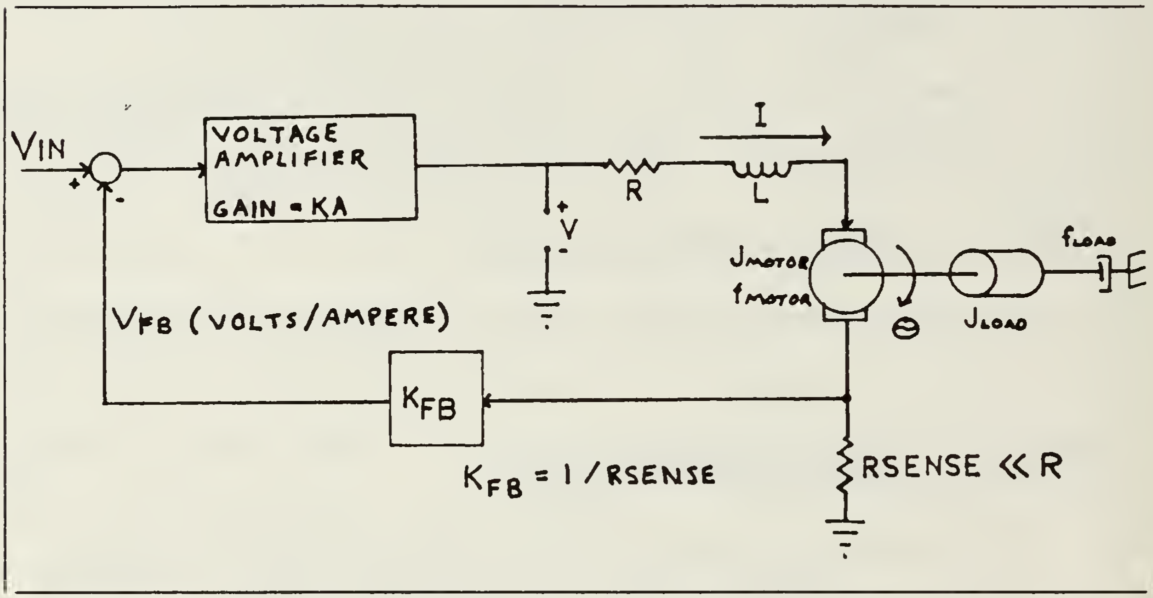

Figure 5.1 Sketch of the Current Source Drive System

Converting Figure 5.1 to a transfer function block diagram and using equations 5.5 and 5.6, the block diagram of Figure 5.2 results. Around the amplifier loop

$$\frac{V}{Vin} = \frac{KA}{1 + KA \dfrac{Js}{[(Js)(R+sL) + Kv\ Kt]}} \qquad (5.8)$$

If the gain of the amplifier is chosen such that

$$KA . \frac{Js}{[(Js)(R+sL) + Kv\ Kt]} >> 1 \qquad (5.9)$$

then

$$\frac{V}{Vin} = \frac{1}{\dfrac{Js}{[(Js)(R+sL) + Kv\ Kt]}} = \frac{[(Js)(R+sL) + Kv\ Kt]}{Js} \quad (5.10)$$

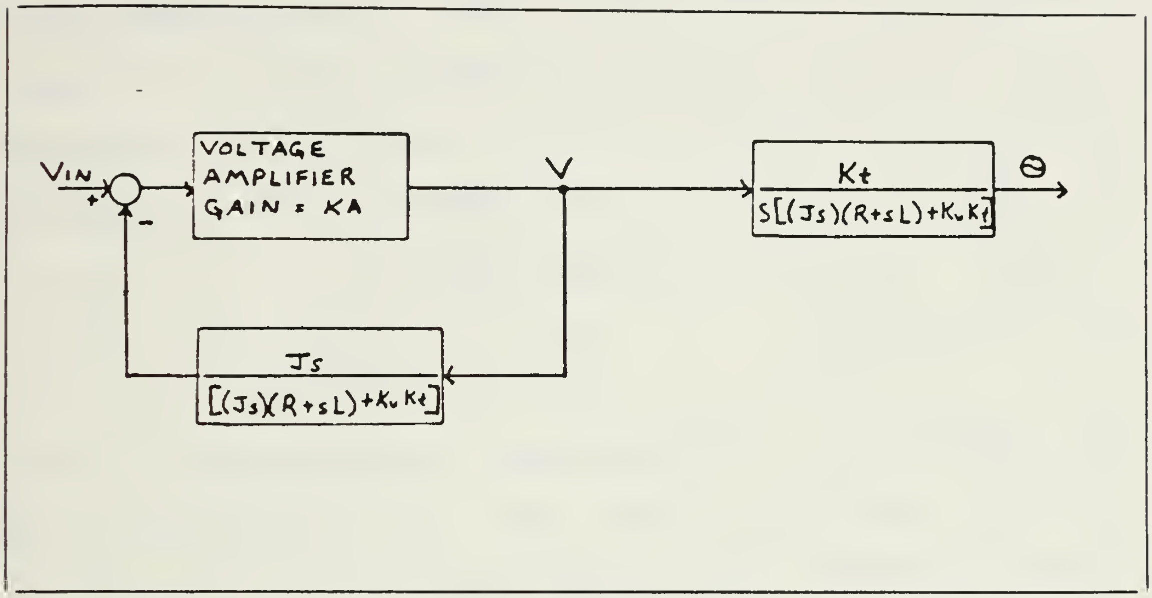

Figure 5.2 Current Source Drive Transfer Function Block Diagram

The equivalent transfer function from Vin to Θ

$$\frac{\Theta(s)}{Vin(s)} = \frac{V}{Vin}\cdot\frac{\Theta}{V} = \frac{[(Js)(R+sL) + Kv\ Kt]}{Js}\,\frac{Kt}{s[(Js)(R+sL) + Kv\ Kt]}$$

$$= \frac{(Kt/J)}{s^2} \quad (5.11)$$

Equation 5.11 indicates that if the amplifier gain is carefully chosen, the armature current feedback yields a system that approximates the ideal motor of the model. The armature inductance and the back emf will have negligible effect on the performance of the motor.

Figure 5.3 is the simulation block diagram of the current source drive system and Appendix C lists the DSL/VS program used to simulate the system. The values used in the

simulation for the servo motor parameters (Kt/J, R, L, and Kv) were obtained from the known poles and gain of the motor transfer function by using the function minimization routine ICSOS where the transfer function of equation 3.2 was run against the transfer function of equation 5.7. The outputs, θ, were minimized for a step command of 0.001 radians. The routine returned the following values

Kt/J	=	275.2,	Kv	=	0.0752
R	=	1.012,	L	=	0.0002862

By substitution of these values into equation 5.7, the transfer function of the servo motor was equal to the original transfer function of equation 3.2.

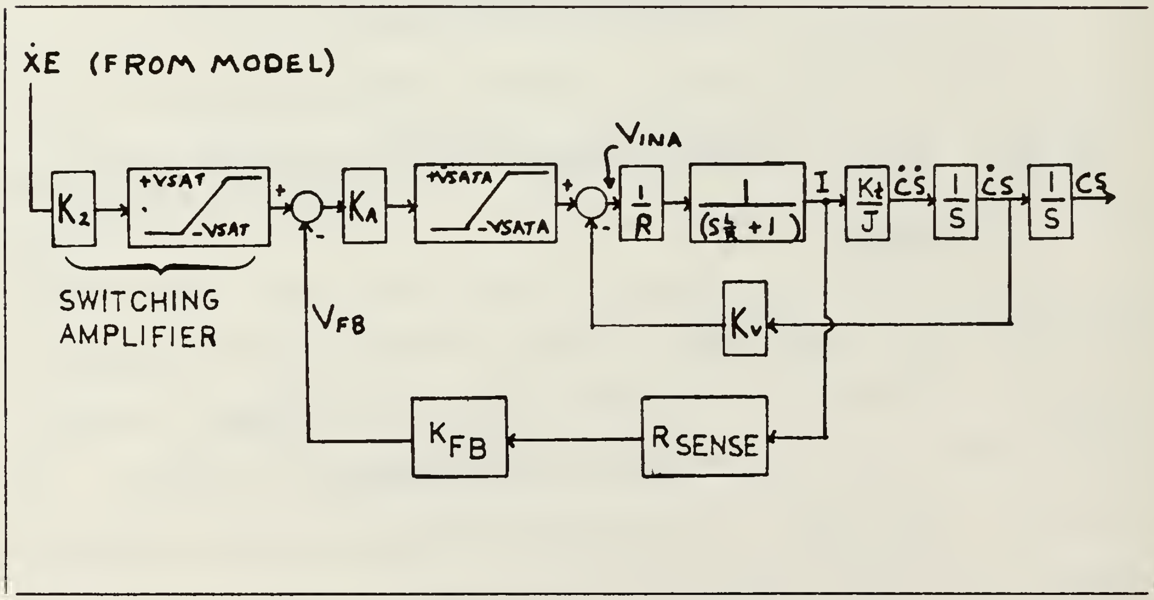

Figure 5.3 Current Source Drive Simulation Block Diagram

RSENSE was chosen to be 0.1 to reduce the influence on the electrical pole, therefore KFB was set at 10.0. The only parameters left to choose were amplifier gain (KA) and

saturation limits (VSATA). This was done by trial and error. KA was initially set to 100 and VSATA was set to a very high value so that no saturation occurred. The trajectory of the current source drive system was almost identical to the model (Figure 5.4) for the one track move. The gain KA was then lowered until curve following was lost (around KA = 10). Then the saturation limit VSATA was lowered and gain KA raised until an acceptable phase trajectory was found that would yield zero head velocity ($\dot{C}S$) when in position. This occurred at KA = 20 and Vsat = 50 volts (Figure 5.5). The $\dot{X}$ curve had to be scaled down (by adjustment of K1 in the model to 0.7 vice 0.8) to switch into curve following early enough to obtain the desired response. The step response of the system is shown in Figure 5.6 and the DSL/VS simulation data listing for this one track move is provided in Appendix C. The data show that the voltage applied to the servo motor VINA is initially 50.0 volts but is rapidlly reduced to around 10.9 volts when the armature current I builds up to 9.79 amperes for the full acceleration portion of the move. When full reverse drive is applied (VS = -10v) VINA is momentarily -50 volts but again dies out quickly as the servo motor follows the curve. This momentary application of + 50 volts allows the current to build rapidly and improved performance is achieved. Because there must be some limitations on the amount of voltage applied to the servo motor, the ideal K/s^2

performance cannot be achieved but this study shows that a practical way of improving the phase trajectory of the 3rd order motor can be obtained with a current source drive system. This study will be continued as the track follow mode is investigated.

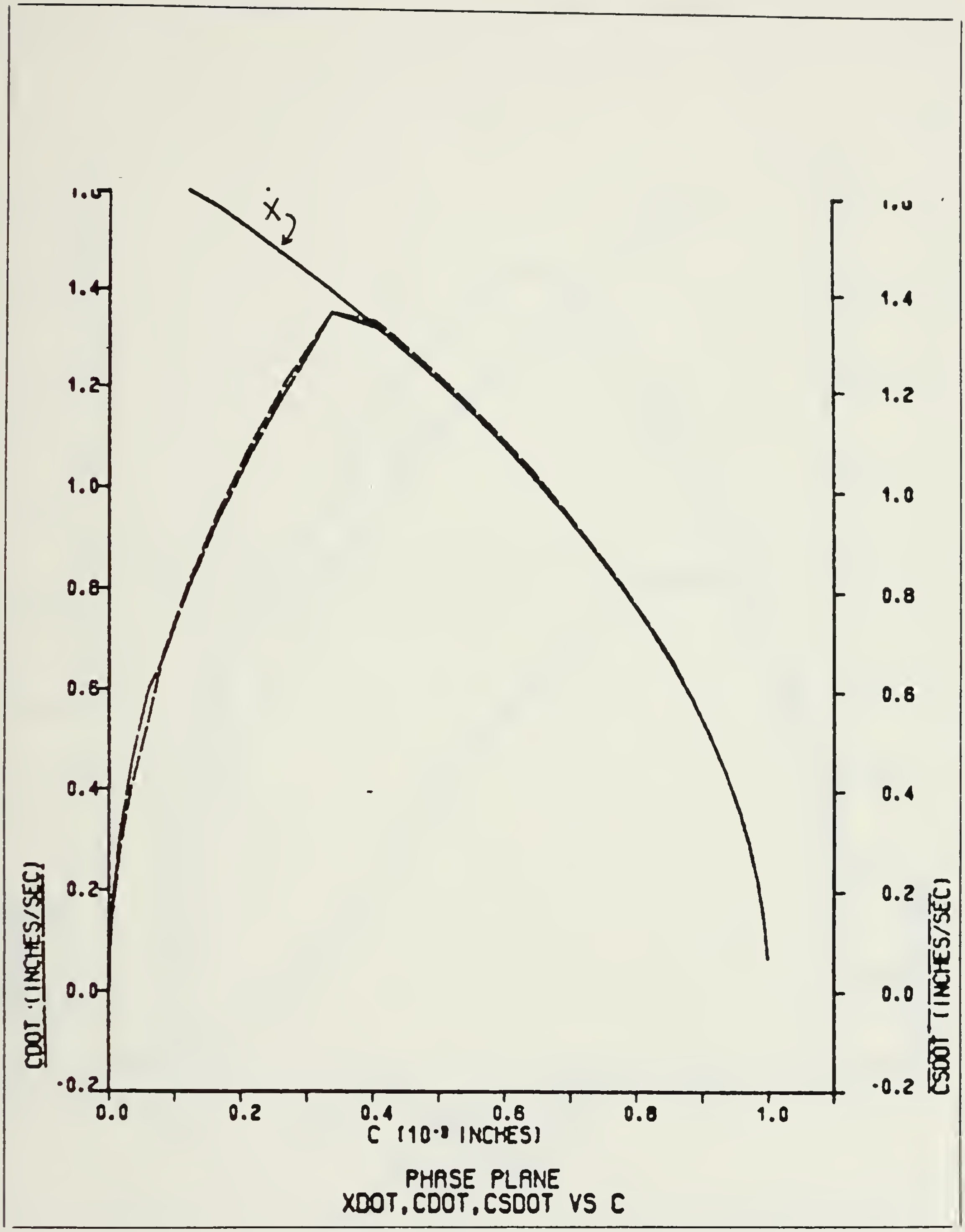

Figure 5.4 Phase Plane Trajectory
Current Source Drive System
KA = 100

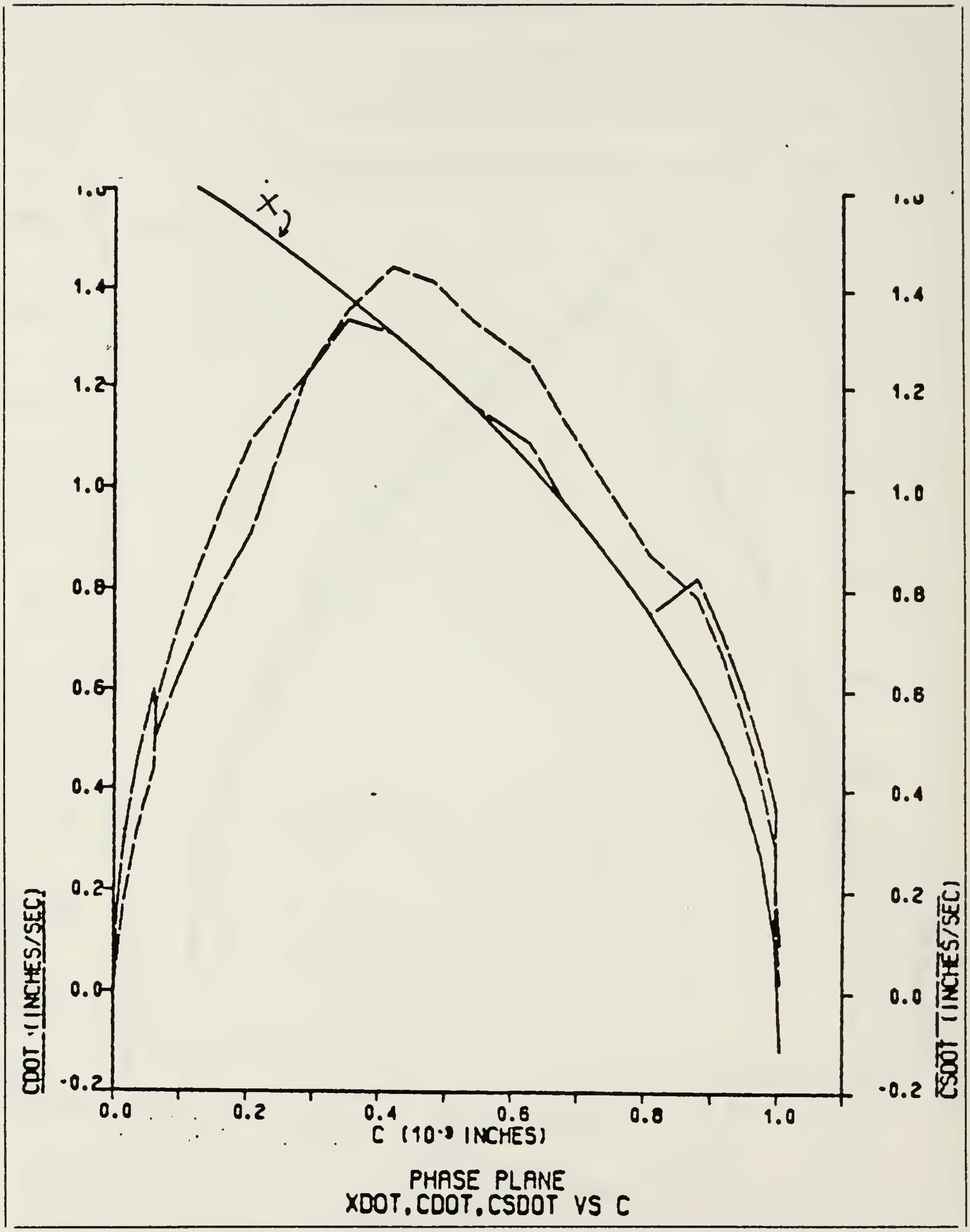

Figure 5.5 Phase Plane Trajectory of
Current Source Drive System
KA = 20, VSATA = 50

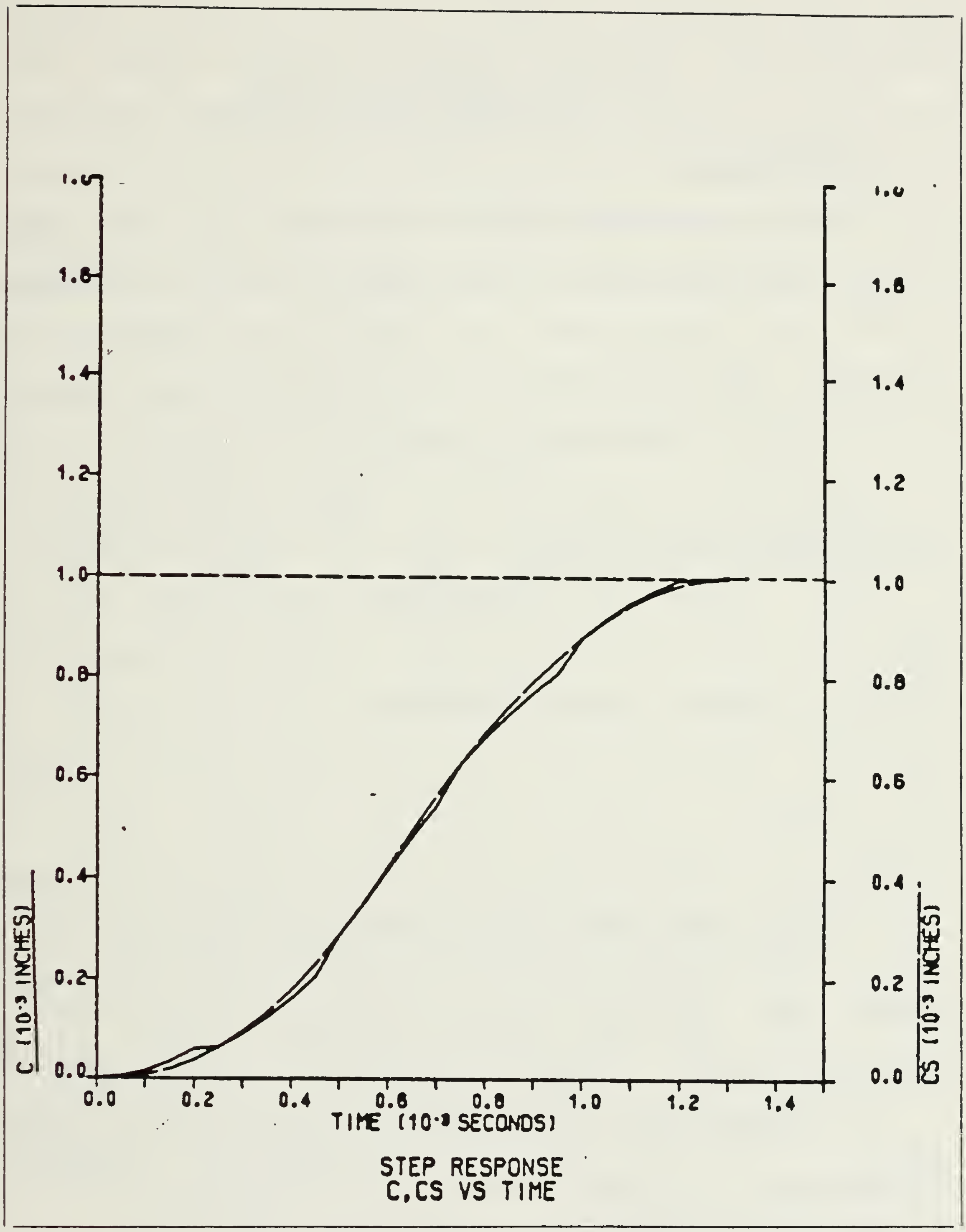

Figure 5.6 Step Response
KA = 20, VSATA = 50

VI. THE TRACK FOLLOW MODE

A. INTRODUCTION

Once the seek mode positions the head over the desired track, the system must switch into the track follow mode to fine tune the head over track center and hold it in this position while the head performs its read/write functions. A well-damped compensation scheme must be utilized to reduce the overshoot of track center and settle any transients in the system in minimum time.

Three methods of accomplishing the track following function will be introduced here. The first scheme is to close a linear compensator around the servo motor and amplifier thereby disconnecting the model from the system. This yields a typical sampled data system. The second method is to replace the curve in the model with the linear compensator and remove the tachometer feedback within the model. The model then drives the servo motor open loop as before with the state and gain parameter updates at the sampling instants. The last method is to replace the curve in the model with a gain block and modify the tachometer feedback gain to provide the required compensation.

B. COMPENSATION CLOSED AROUND THE SERVO

Figure 6.1 is the block diagram of the sampled data compensation scheme for the track follow mode where the

samples of head position are held by use of a zero order hold circuit between sampling instants. The nonlinear saturating amplifier can be assumed to operate in the linear region when in the track follow mode if the head is over, or very close to, track center when switching occurs. This assumption allows linear analysis to be performed on the closed loop.

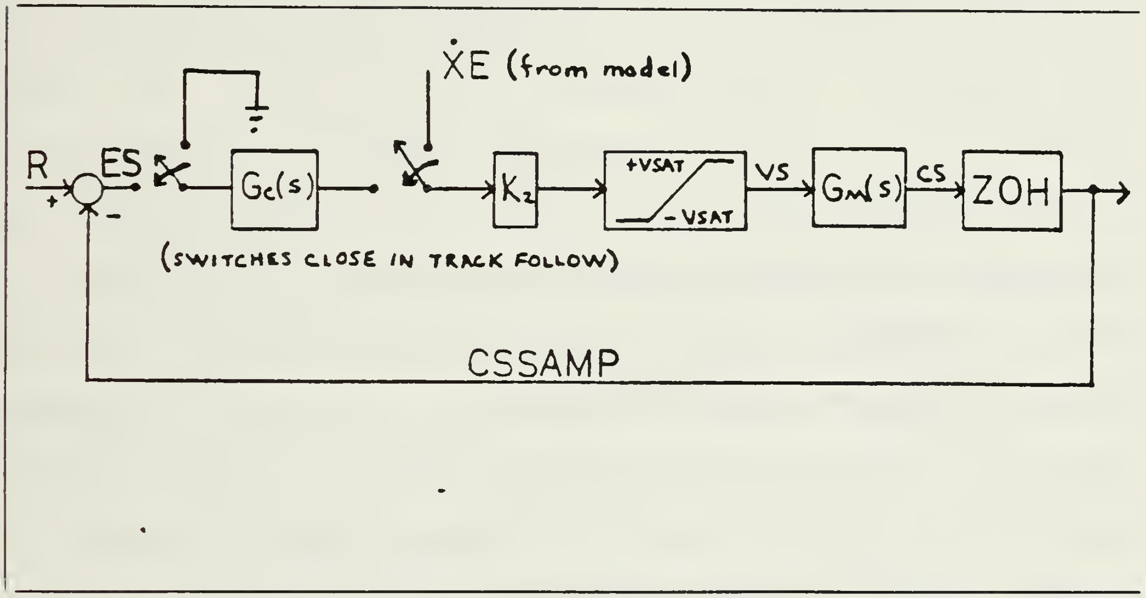

Figure 6.1 Block Diagram of the Track Follow Mode for Compensation Closed Around the Servo

The samples of head position are known at the sampling instants NT. By holding the values of head position, the output of the zero order hold can be represented by

$$CSSAMP = \frac{1 - e^{ST}}{s} \cdot CS \tag{6.1}$$

The order zero hold circuit can have a destabilizing effect on the closed loop system because it introduces a lagging phase angle and attenuation in the forward path. This

effect is best seen by examining the frequency response of the zero order hold [Ref. 3: pp 80-82] where

$$Gzoh\ (s) = \frac{\sin(\pi\, w/ws)}{w/ws}\ e^{-j(\pi w/ws)t} \tag{6.2}$$

$$|Gzoh\ (s)| = \left|\frac{\sin(\pi\, w/ws)}{w/ws}\right| \tag{6.3}$$

$$\underline{\angle Gzoh\ (s)} = \frac{-\pi w}{ws}\ , \quad \frac{\sin(\pi\, w/ws)}{w/ws} > 0 \tag{6.4}$$

In this analysis, w is the test frequency and ws is the sampling frequency. If the sampling period is 0.25 milliseconds (approximately 64 sectors of servo information), the sampling frequency is 25,133 radians/second. In most disk files the bandwidth of the compensated system is approximately 3000 radians/second. At this bandwidth, the zero order hold introduces 21.5 of phase lag and -0.2dB of attenuation. Thus the compensator must be chosen such that it has sufficient phase lead to overcome the influence of the zero order hold circuit.

The Gm(s) transfer function block in Figure 6.1 can be either that of the second order servo motor (equation 3.3) or $(Kt/J)/s^2$ if the current source drive system is used as they will have similar frequency responses in the compensated system. The compensator chosen to provide the required phase lead was a single section phase lead filter with transfer function

$$Gc(s) = \frac{\dfrac{s}{1000} + 1}{\dfrac{s}{50,000} + 1} \tag{6.5}$$

The open loop Bode plot of the compensated system considering only Gc(s), K2 , and Gm(s) is shown in Figure 6.2. This plot shows the phase margin to be greater than 70° with a bandwidth of approximately 2900 radians/second. The phase lead is large enough to overcome the influence of the zero order hold and provide a well damped response. Simulation studies will be presented at the end of this chapter where the performance of the three methods of compensation will be compared.

C. COMPENSATION FILTER IN THE MODEL

For this method of compensation, the filter of equation 6.5 replaces the curve in the model as shown in Figure 6.3. An open loop Bode plot of the compensated model would be similar to Figure 6.2 when the curve is replaced by the filter and the tachometer feedback is removed (K = 0.0).

Again, the transfer function Gm(s) is either the transfer function of the second order servo motor or the current source drive system. The microprocessor would change the value of K from 1.0 to 0.0 when switching to track follow and replace the curve with the filter to compensate the model. The servo motor is driven open loop for both the seek mode and track follow mode with updates of model states and motor gain parameter Km provided at the sampling instants. Thus the system is not a sampled data system as in Section B above.

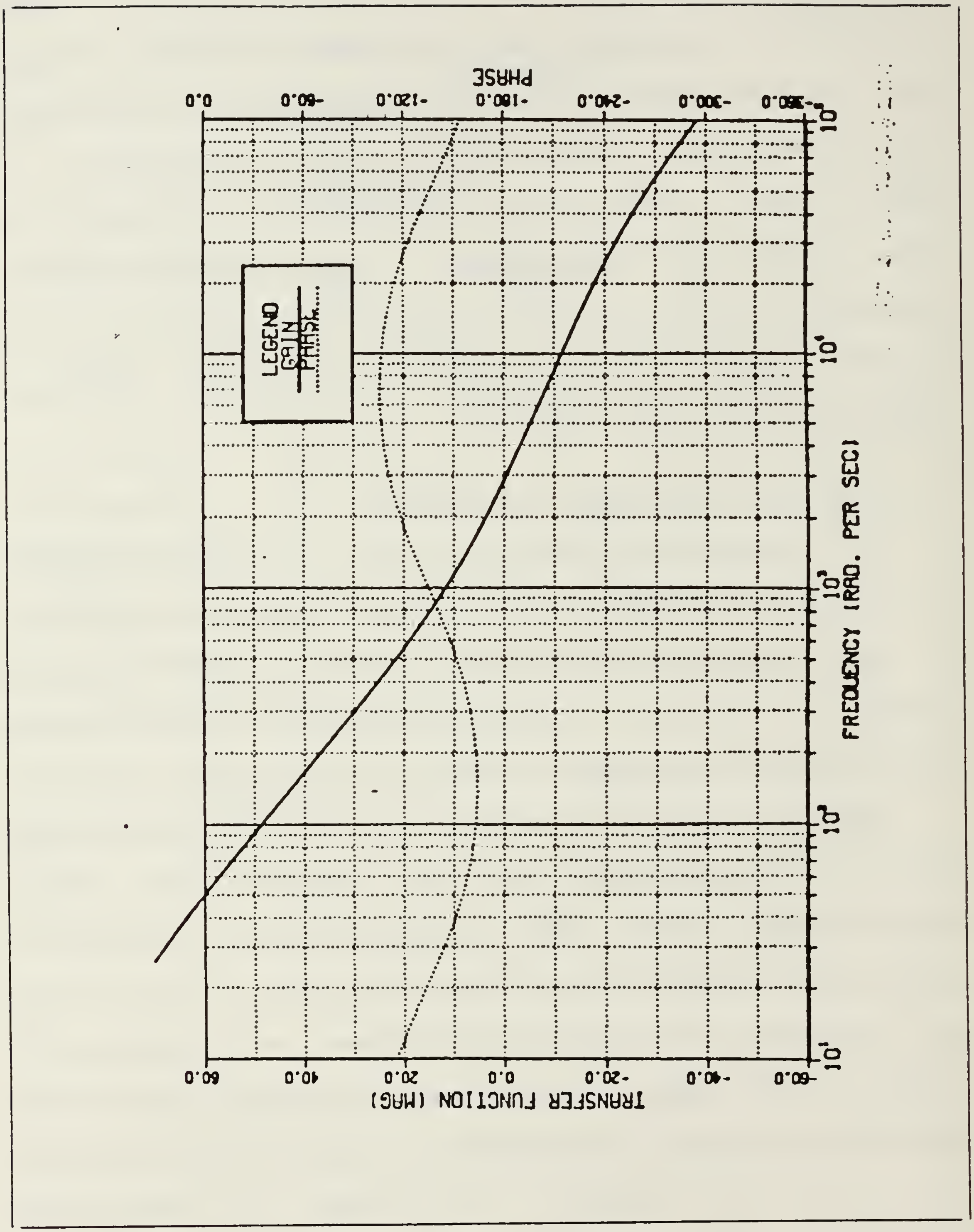

Figure 6.2 Open Loop Bode Plot of Compensated System for the Track Follow Mode

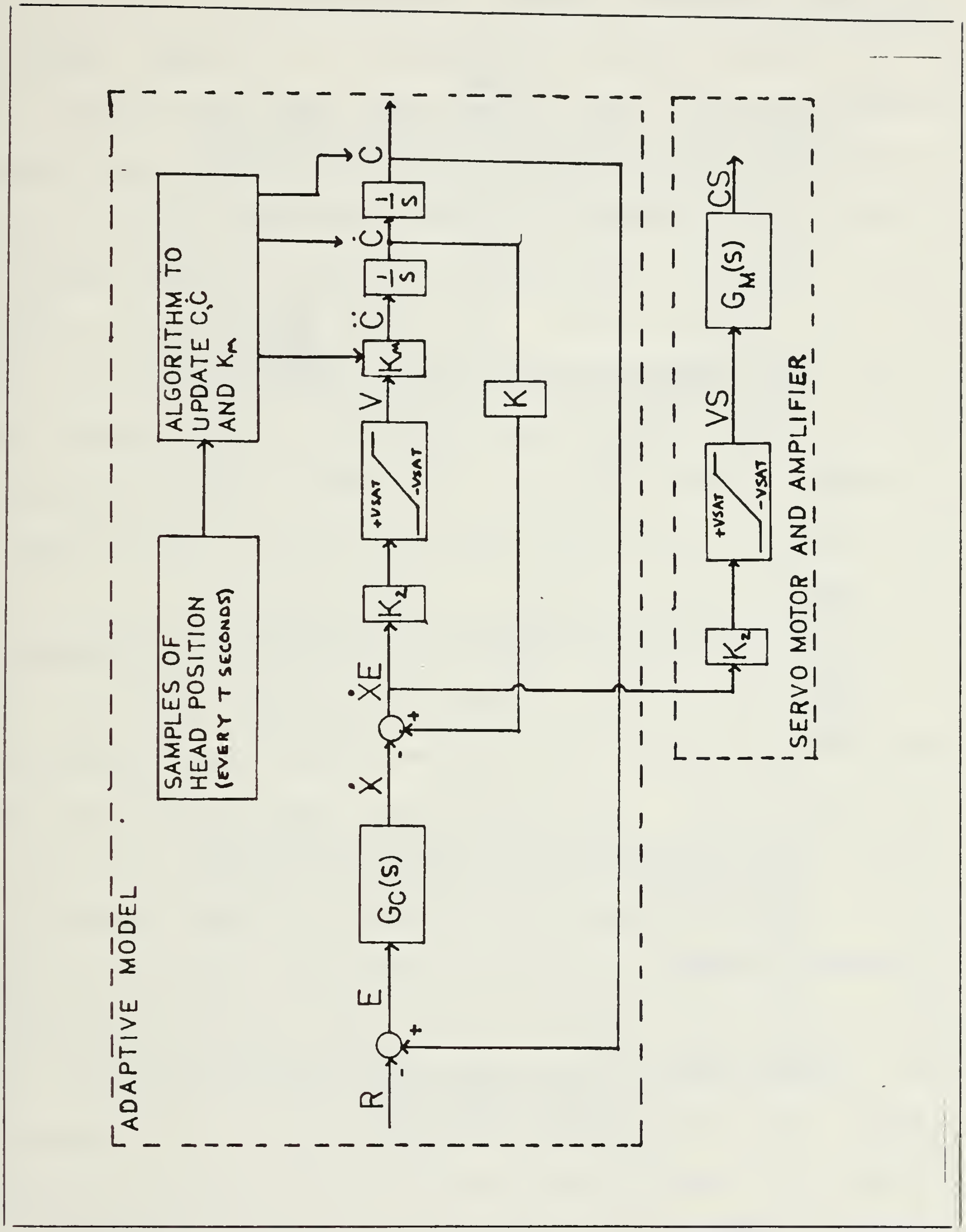

Figure 6.3 Compensation Filter Placed in the Model for Track Follow Mode

D. MODEL TACHOMETER FEEDBACK COMPENSATION

For this method of compensation, the model curve is replaced with a gain compensation block Gc(s) = Ke and the tachometer gain (K) is adjusted to provide a well-damped model in the track follow mode. By Bode analysis of the open loop model, the values for the gain (Ke) was found to be 32.0 with K = 0.01. As figure 6.4 shows, the phase margin is approximately 80° with a gain crossover of 3000 radians/second for the compensated model using the tachometer feedback compensation scheme. A block diagram of this scheme would be the same as Figure 6.3 with Gc(s) = 32.0 and K = 0.01.

E. SIMULATION STUDIES FOR THE THREE METHODS OF COMPENSATION

Appendix D lists the DSL/VS programs used in the simulation studies for the three methods of compensation discussed above. The decision was made to switch into the track follow mode when the head was over track center (E = 0) in order to compare the performance of each method of compensation under the same switching law. Figures 6.5 and 6.6 are the step response curves for a one track move when the compensation filter is closed around the servo for the second order servo motor and current source drive systems. Because the model is disconnected in the track follow mode, its step response (C vs time) is not plotted. Figures 6.7 and 6.8 are the step response curves for the one track move with the compensation filter placed in the model and model

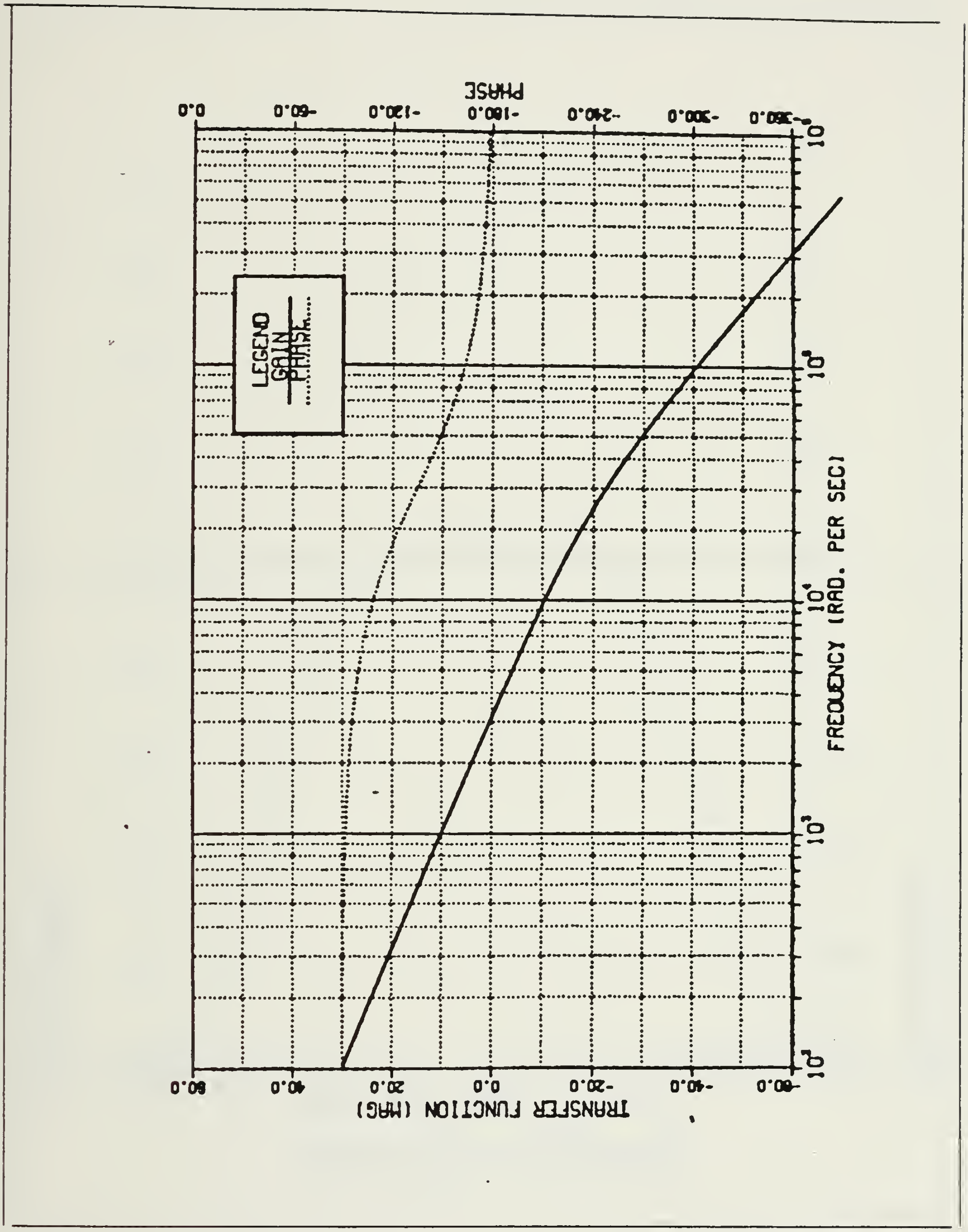

Figure 6.4 Open Loop Bode Plot of the Compensated Model with Tachometer Feedback Compensation (K = 32.0, k = 0.1)

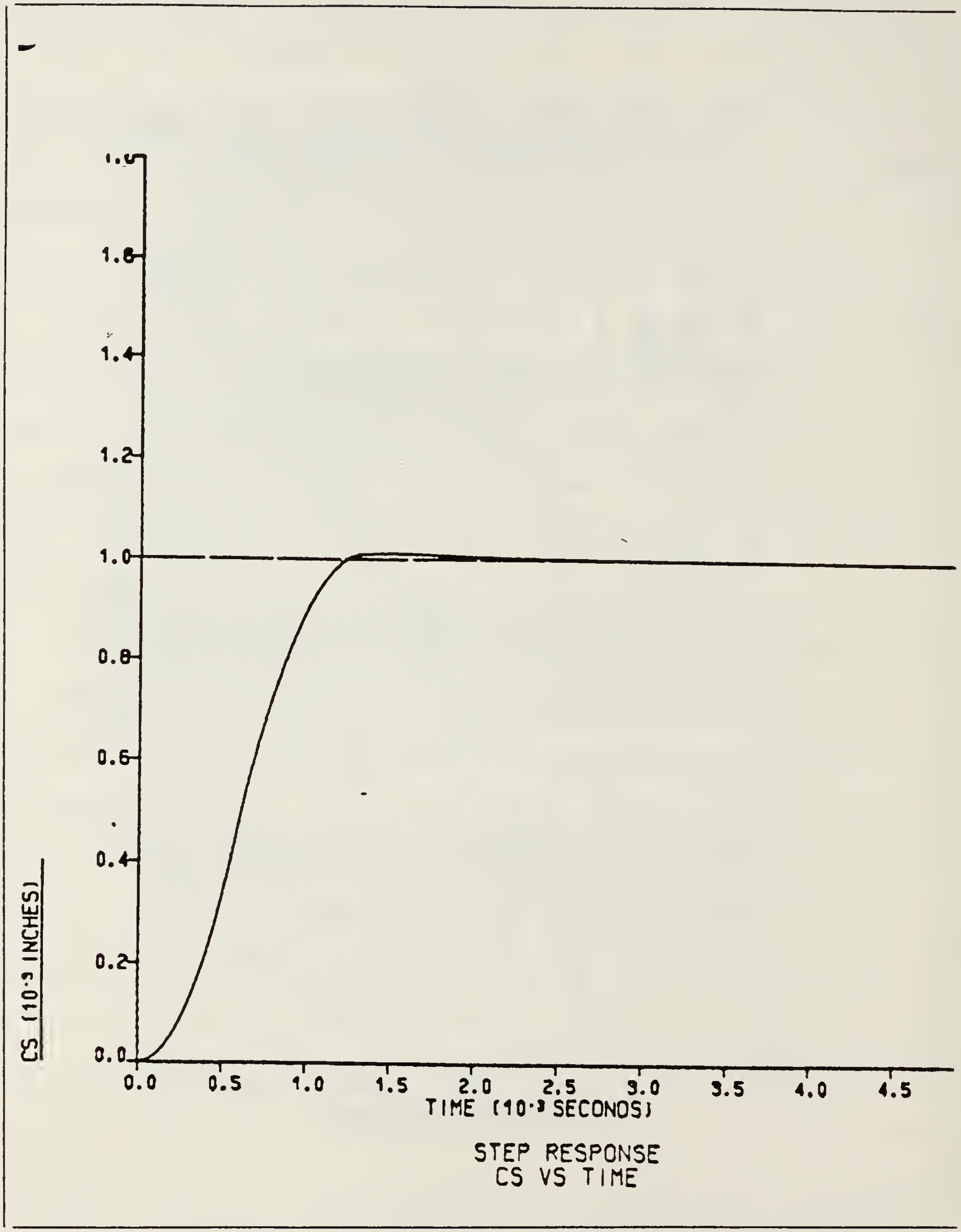

Figure 6.5 Compensation Closed Around the Servo
(Second Order Motor)

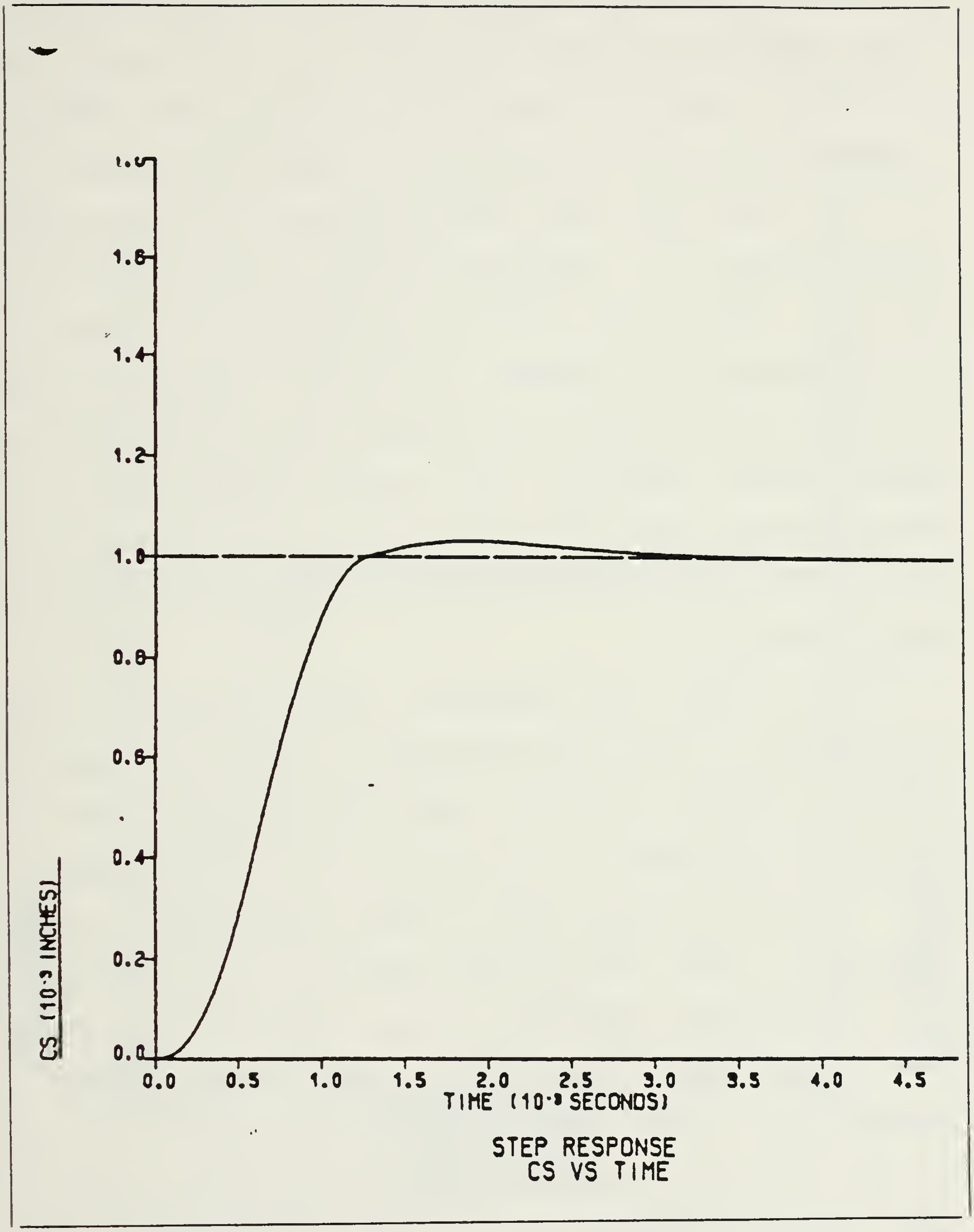

Figure 6.6 Compensation Closed Around Servo
(Current Source Drive)

tachometer feedback removed. Figures 6.9 and 6.10 are the step response curves for the one track move using model tachometer feedback compensation. During the simulation studies of the model compensation schemes (sections 6.C and 6.D) it was found that when the average velocity updates were provided to the model ($\dot{C}S$ avg of equation 3.7) the settling time of the system was significantly reduced and thus Figures 6.7 through 6.10 are the result of using average velocity updates during the track follow mode. This would entail switching the velocity algorithms in the microprocessor when going from seek to track follow and will reduce the amount of calculations required to update the model velocity.

If the head can read/write without adjacent track interference (noise) when the head is within $\pm$ 5% of track center (50 micro inches for track widths of .001 inch) the head can be assumed to be in position at the beginning of the track follow mode for all three compensation methods presented. Thus the settling time for a one track move is 1.3 milliseconds regardless of the type of compensation used. Exhaustive simulation runs for each type of compensation showed that for any length move (1 to 1000 tracks) all three compensation schemes were able to hold the head position to within $\pm$ 5% of track center from the start of the track follow mode. As the length of the move increased, the overshoot (or undershoot) of track center was

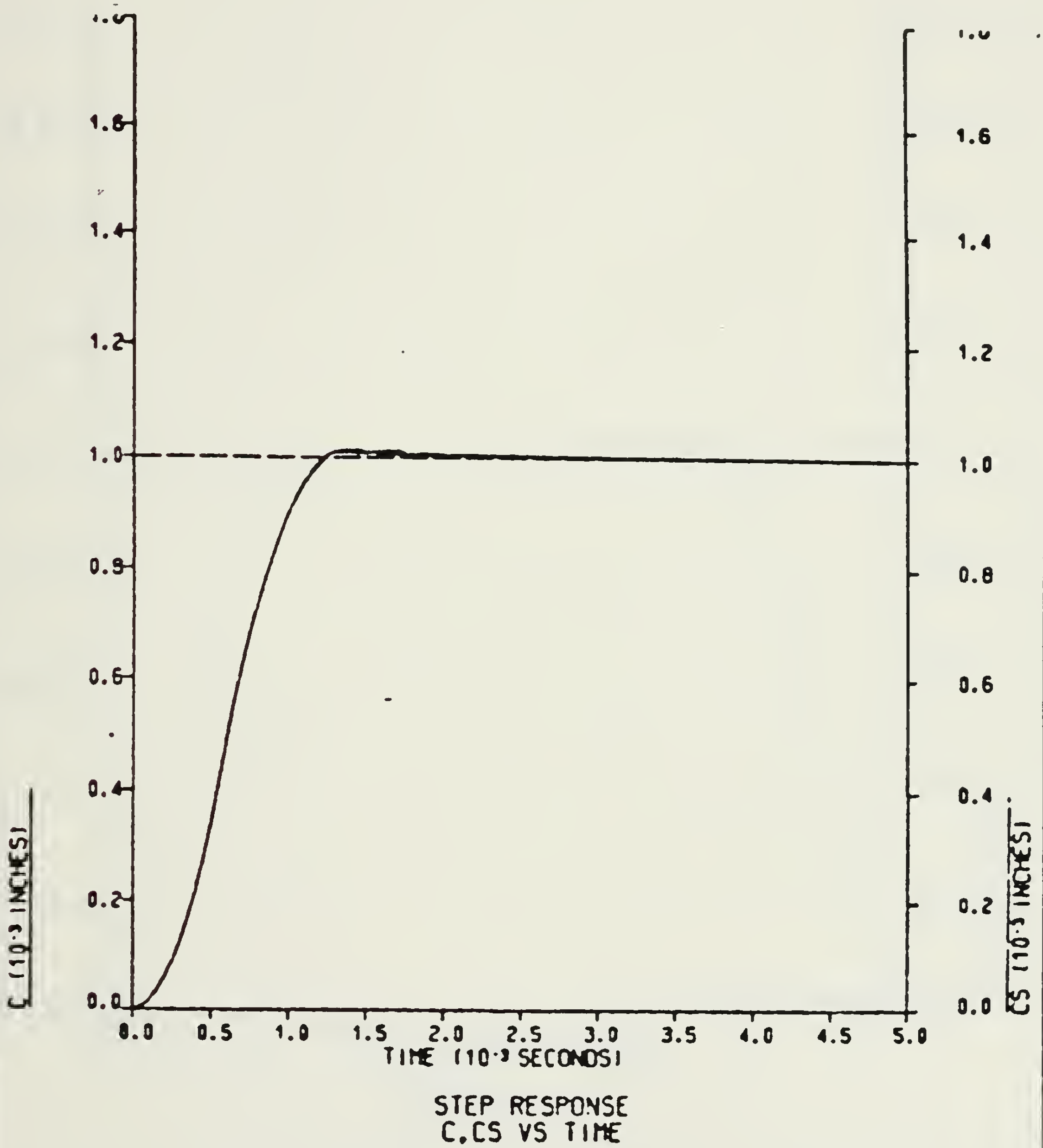

Figure 6.7 Compensation Filter in Model
and Tach Feedback Removed
(Second Order Motor)

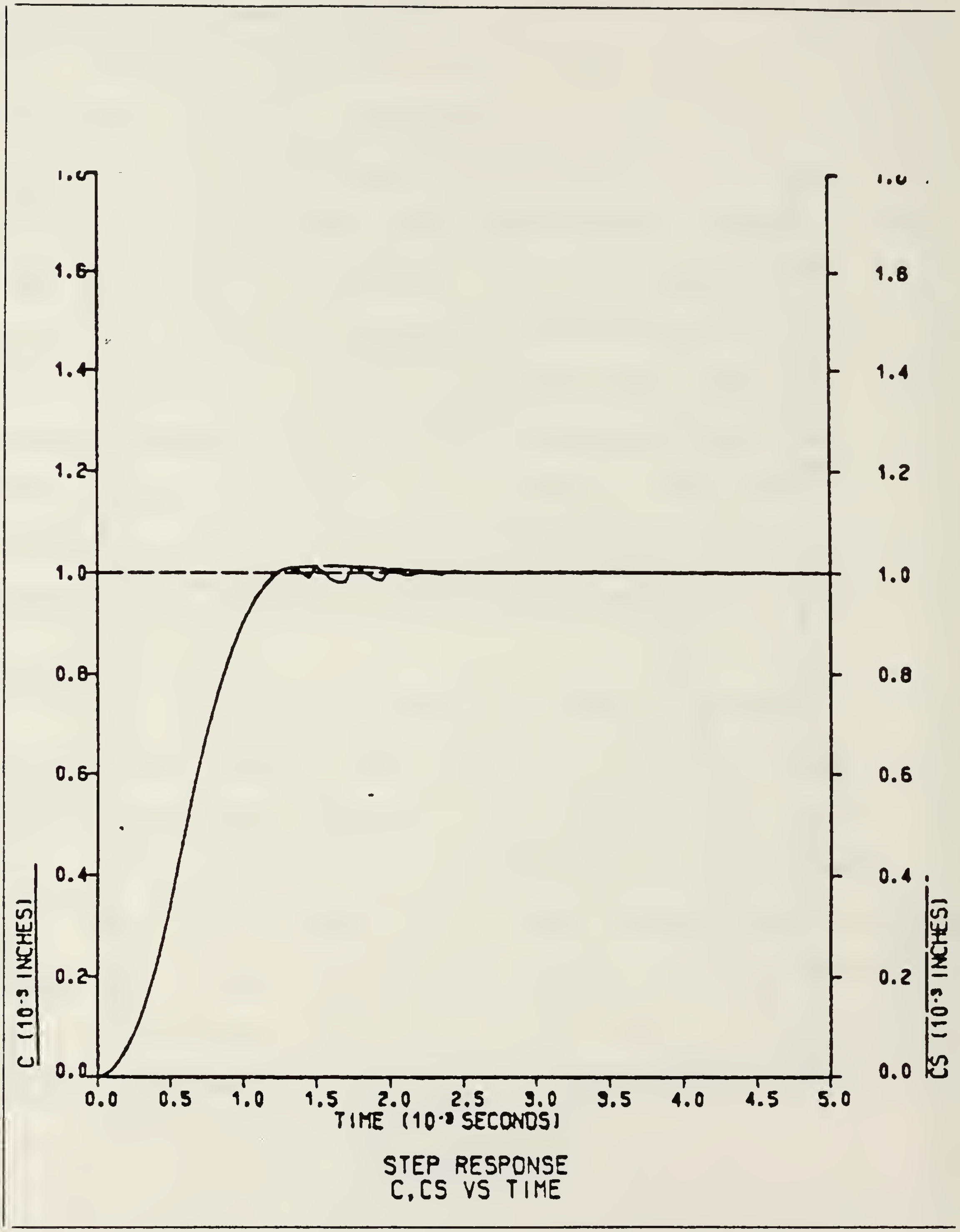

Figure 6.8 Compensation Filter in Model
and Tach Feedback Removed
(Current Source Drive System)

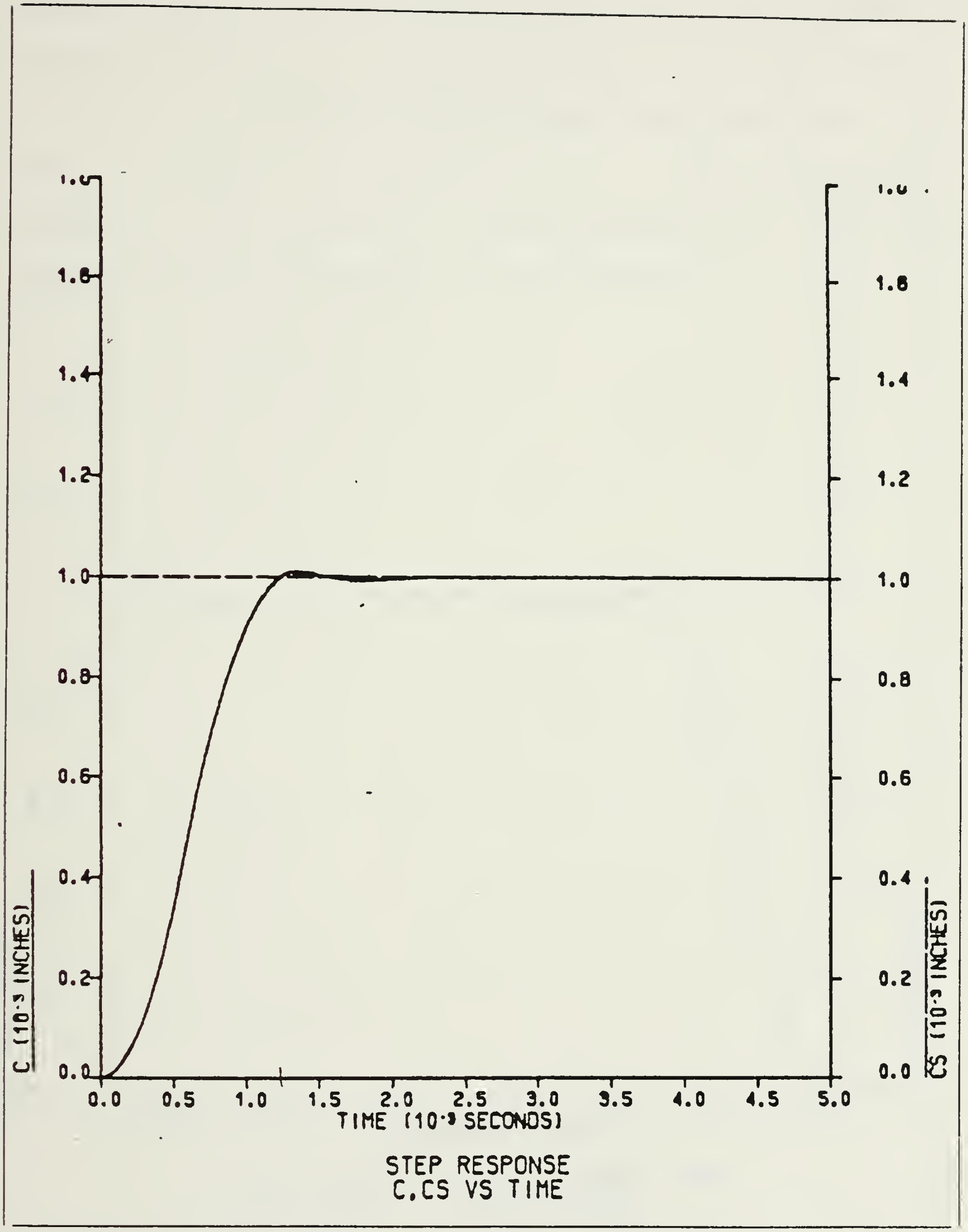

Figure 6.9 Model Tachometer Feedback Compensation
(Second Order Motor)

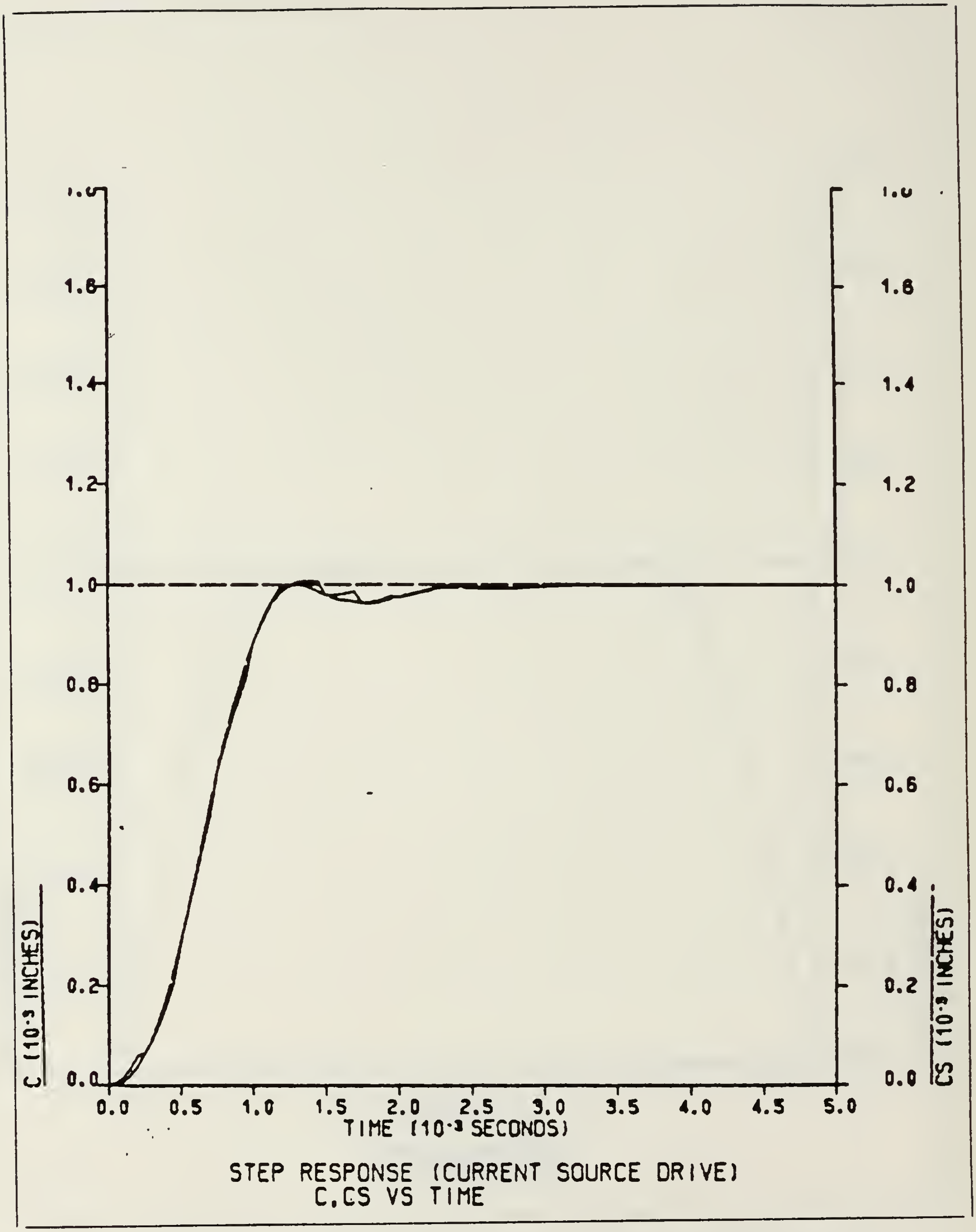

Figure 6.10 Model Tachometer Feedback Compensation (Current Source Drive)

reduced, and in all cases, the one track moves shown in Figures 6.5 through 6.10 were the worst cases encountered.

In the next chapter, a compensation scheme will be chosen based on ease of implementation into a disk file system. The effects of varying the sampling period and introducing computation time delays will also be examined.

VII. THE COMPLETE SYSTEM

A. INTRODUCTION

In the preceeding chapter, three methods of compensating the track follow mode were presented and all were shown to quickly settle the transients of the servo motor. This chapter will first consider the merits of each type of compensation and select a scheme for possible implementation into microprocessor. For the chosen system, a complete set of figures showing the phase trajectory, step response, servo motor drive signals and velocity of the servo motor will be presented. Finally the effects of varying the sampling period and introcuding time delays into the system will be investigated.

B. CHOOSING THE BEST COMPENSATION SCHEME FOR THE TRACK FOLLOW MODE

When compensation is closed around the servo motor, the model is switched out of the system. The compensation can be provided by utilizing an analog or digital filter implemented in hardware thereby releasing the microprocessor for other tasks until the next move. Another method would be to implement a digital filter algorithm in software. While the choice would be left to the system designer, maximum use of the microprocessor during the seek and track follow modes by using a software implemented filter may be

the best choice. One drawback to consider is the effect of the sampling rate on the system. If the sampling rate is decreased too far, the zero order hold circuit may introduce enough phase lag to destabilize the closed loop.

When the compensation filter replaces the curve in the model, updates of the servo motor states are still required. The implementation of the filter as a discrete algorithm may increase the computations required while track following. The zero order hold is not used in this system and the effects of sampling rate should not be a critical factor in the stability of the system.

If compensation is provided by changing the tachometer feedback of the model and replacing the curve with gain compensation, the computation load on the microprocessor is reduced for track following. If the curve was generated by table look up the reading of commanded velocity values ($\dot{X}$) from a ROM is eliminated. Again, model states are updated with this scheme and the sampling rate may not be as critical as in the first method discussed.

Based on the observation above, the tachometer feedback compensation of the model may be the best choice for the track follow mode. The simulation model is used for both the seek and track follow modes and the algorithm for each is basically the same. This scheme would allow maximum utilization of the microprocessor in a system design.

C. SIMULATION PLOTS FOR THE COMPLETE SYSTEM

A complete set of figures showing the system response for both the seek and track follow modes are presented here. Model tachometer feedback is used for the track follow mode and current source drive is applied to the servo motor to illustrate the motor armature current. Figures 7.1 through 7.5 show the one track move phase trajectory, step response, servo motor drive voltage (VS), armature current and velocity profile of the servo motor. Figures 7.6 through 7.10 are similar plots for the 100 track move. Figures 7.8 and 7.9 are of particular interest in that they show how the armature current builds quickly when full drive is applied during the acceleration portion of the seek mode and how the drive signal switches between forward and reverse drive as the servo curve follows or "chatters down" the curve. When the compensation is applied for track following, the servo motor drive signal and armature current are seen to decrease to zero rapidly indicating that the head is finely positioned over track center. Figure 7.10 illustrates the effectiveness of the model curve following velocity loop to produce a practical bang-bang control system. Constant acceleration and deceleration is observed for the seek mode and when switching to the track follow mode occurs, the head velocity will be at a minimum. The response curves of the second order motor (with voltage source drive) are almost identical to the figures shown and will not be presented

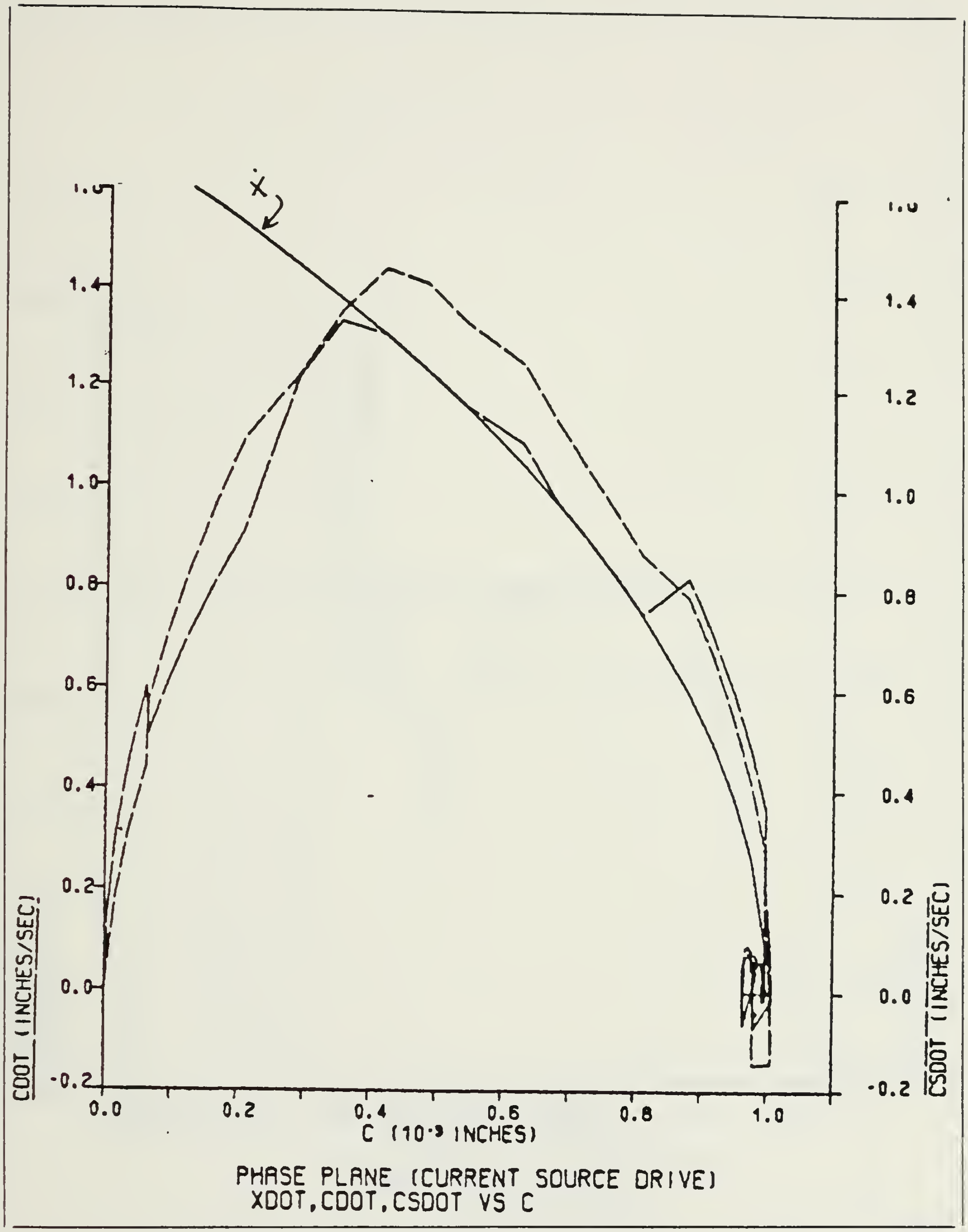

Figure 7.1 Phase Trajectory (1 Track Move)

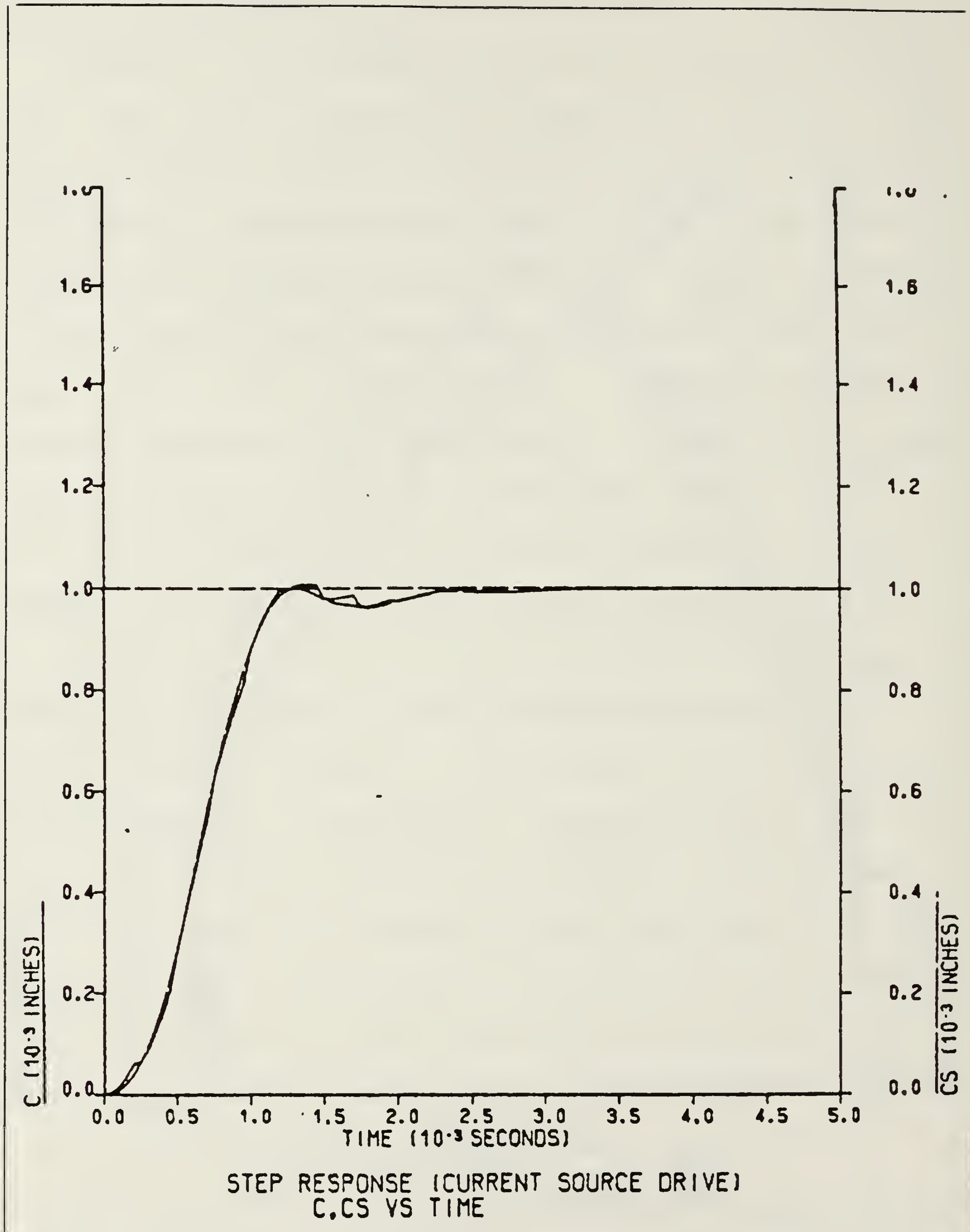

Figure 7.2 Step Response (1 Track Move)

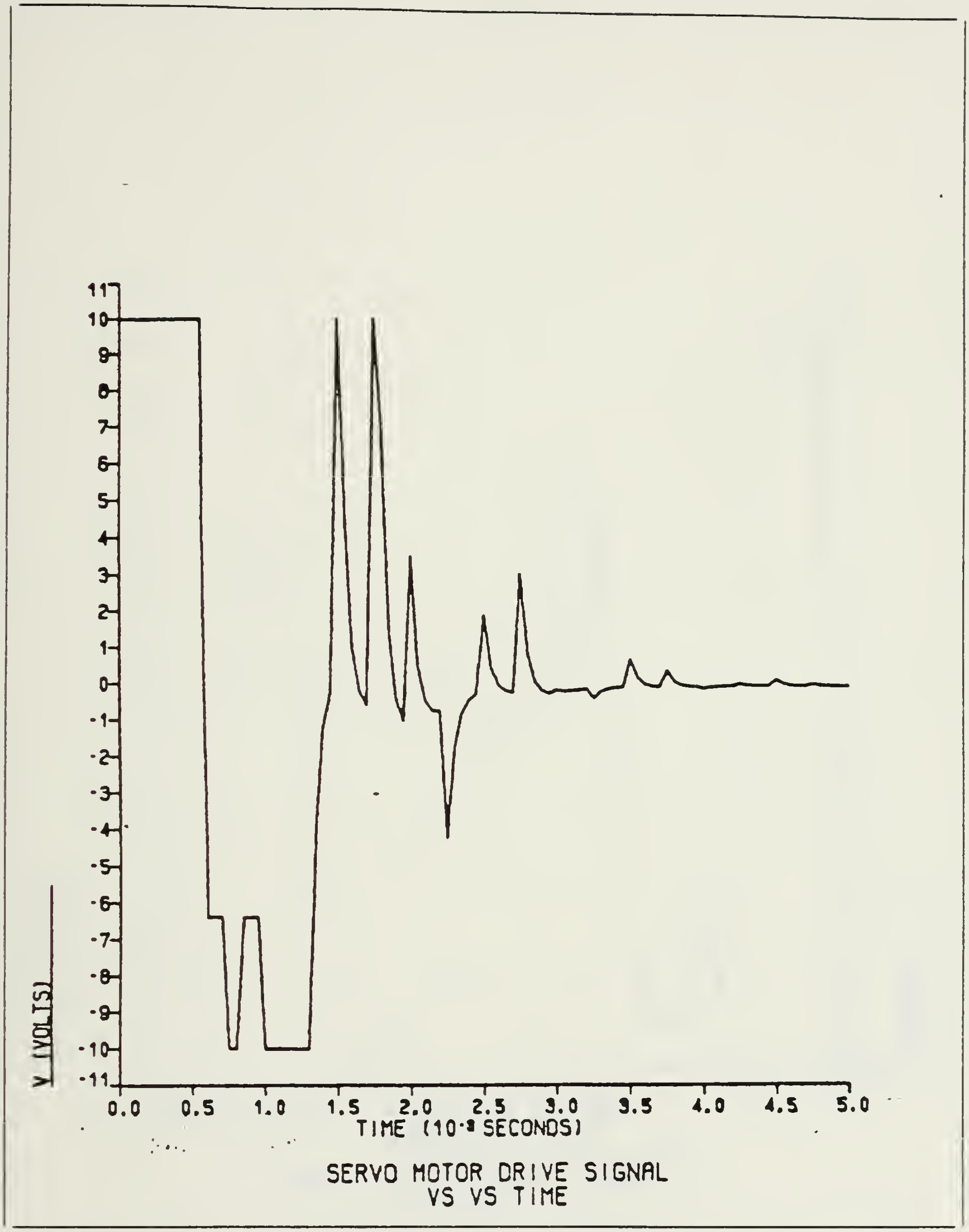

Figure 7.3 Servo Motor Drive Voltage
(1 Track Move)

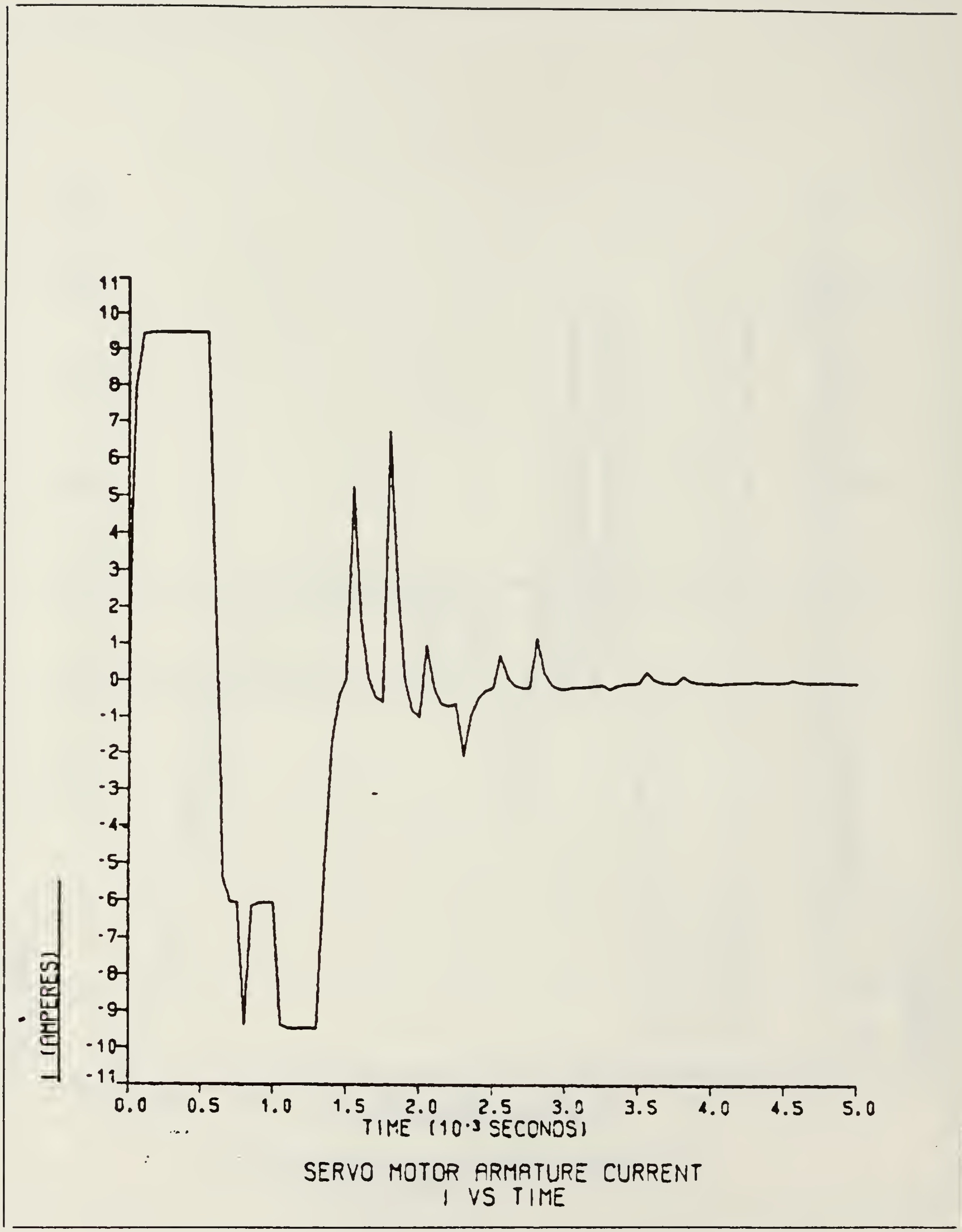

Figure 7.4 Servo Motor Armature Current
(1 Track Move)

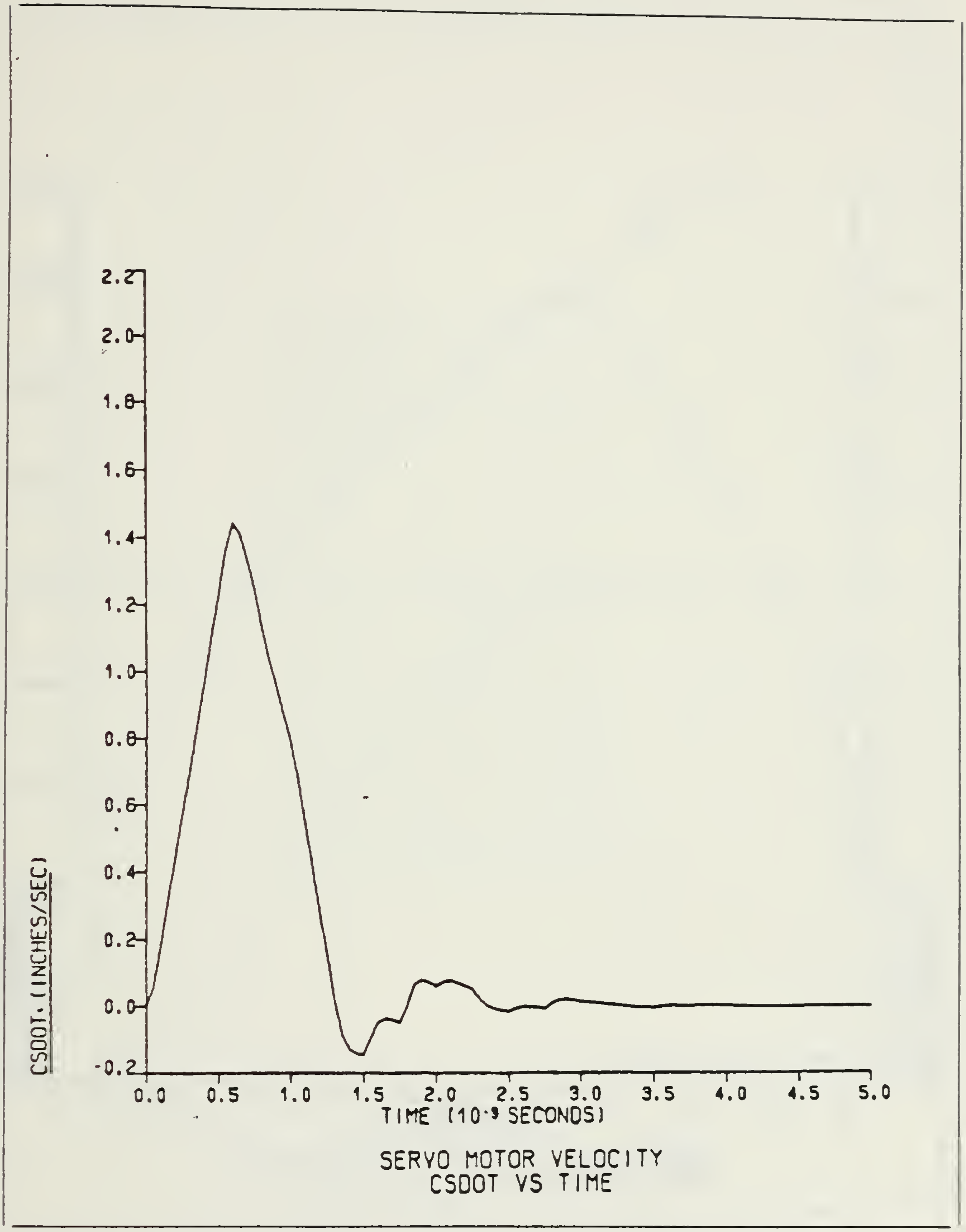

Figure 7.5 Servo Motor Velocity
(1 Track Move)

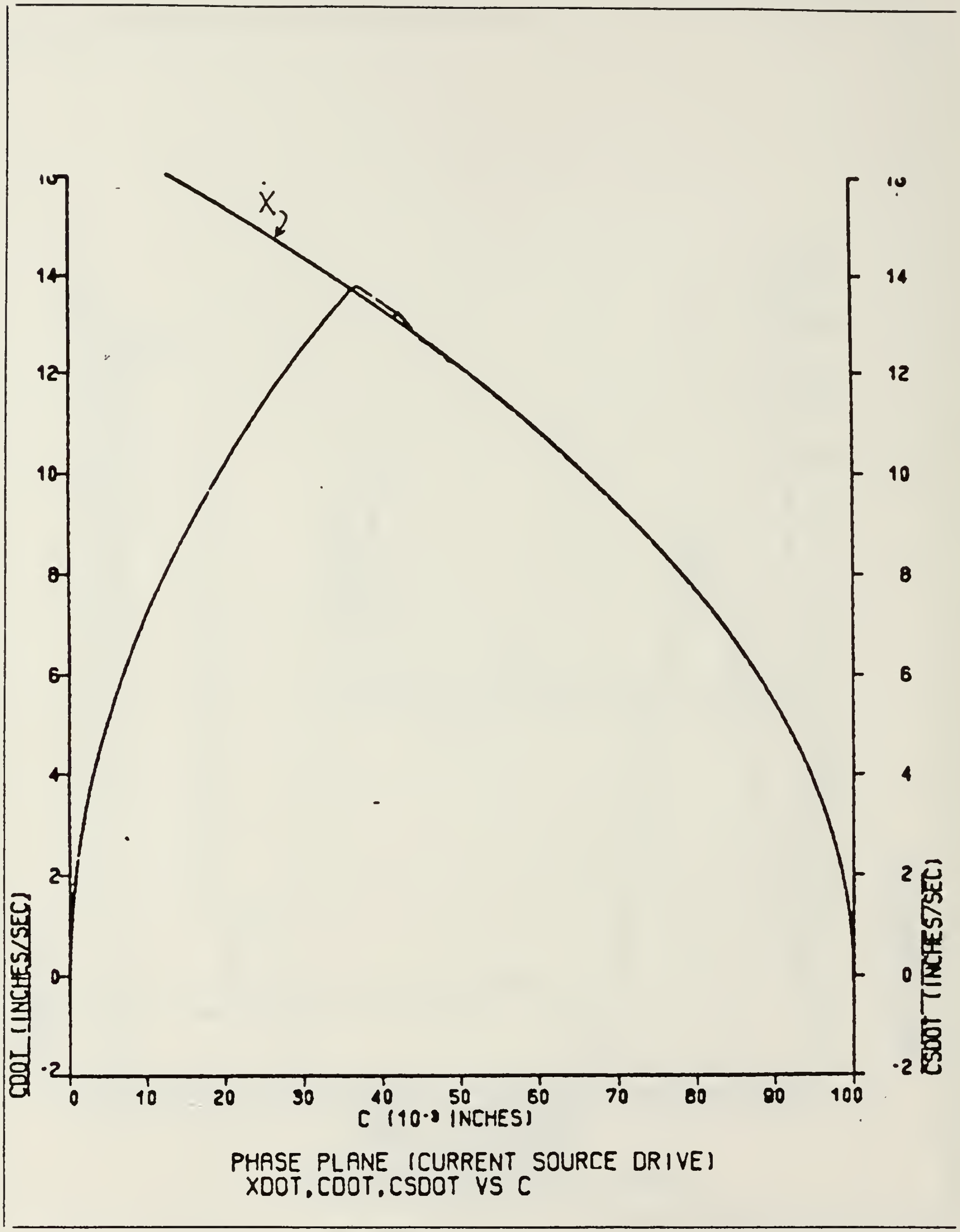

Figure 7.6 Phase Trajectory
(100 Track Move)

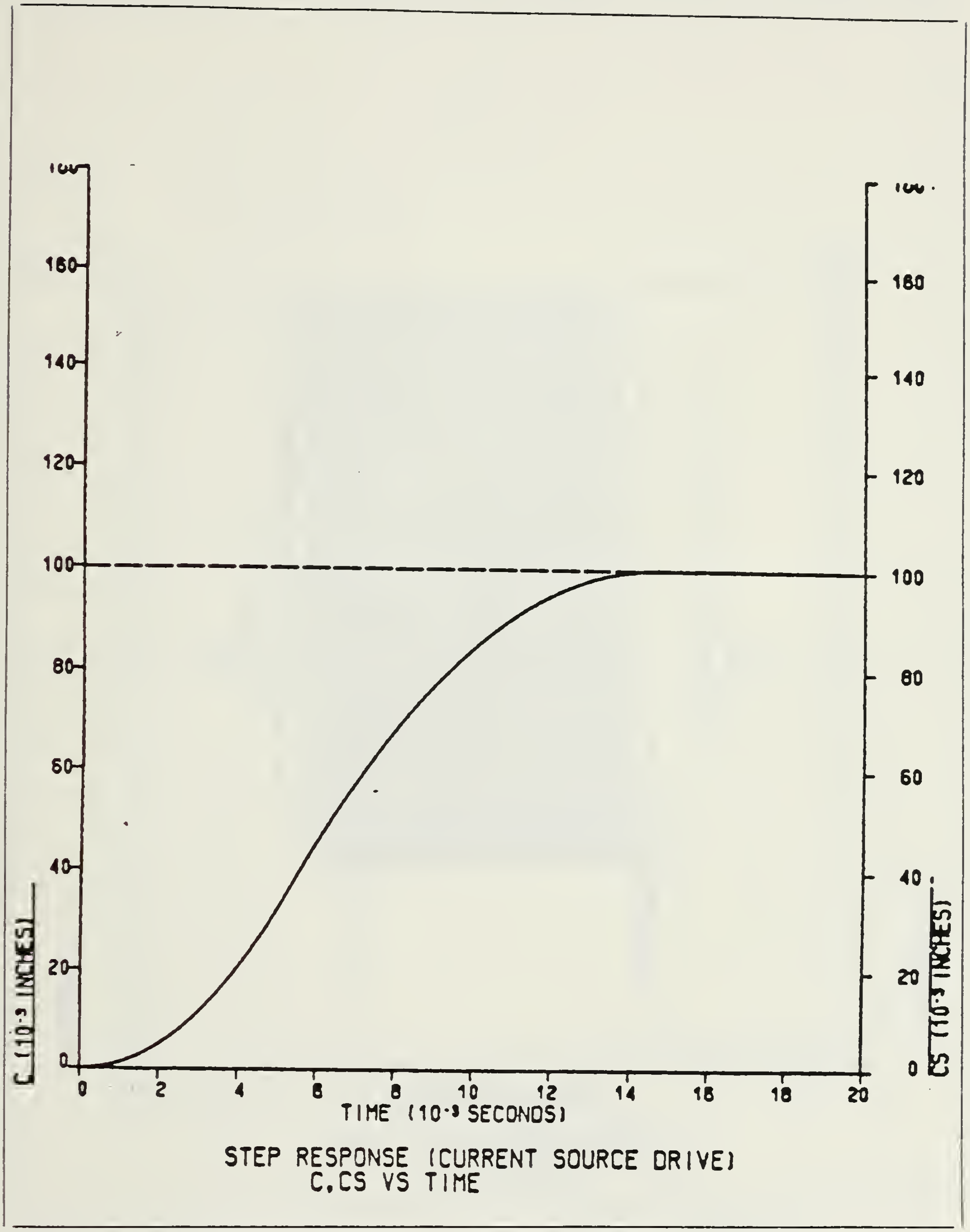

Figure 7.7 Step Response
(100 Track Move)

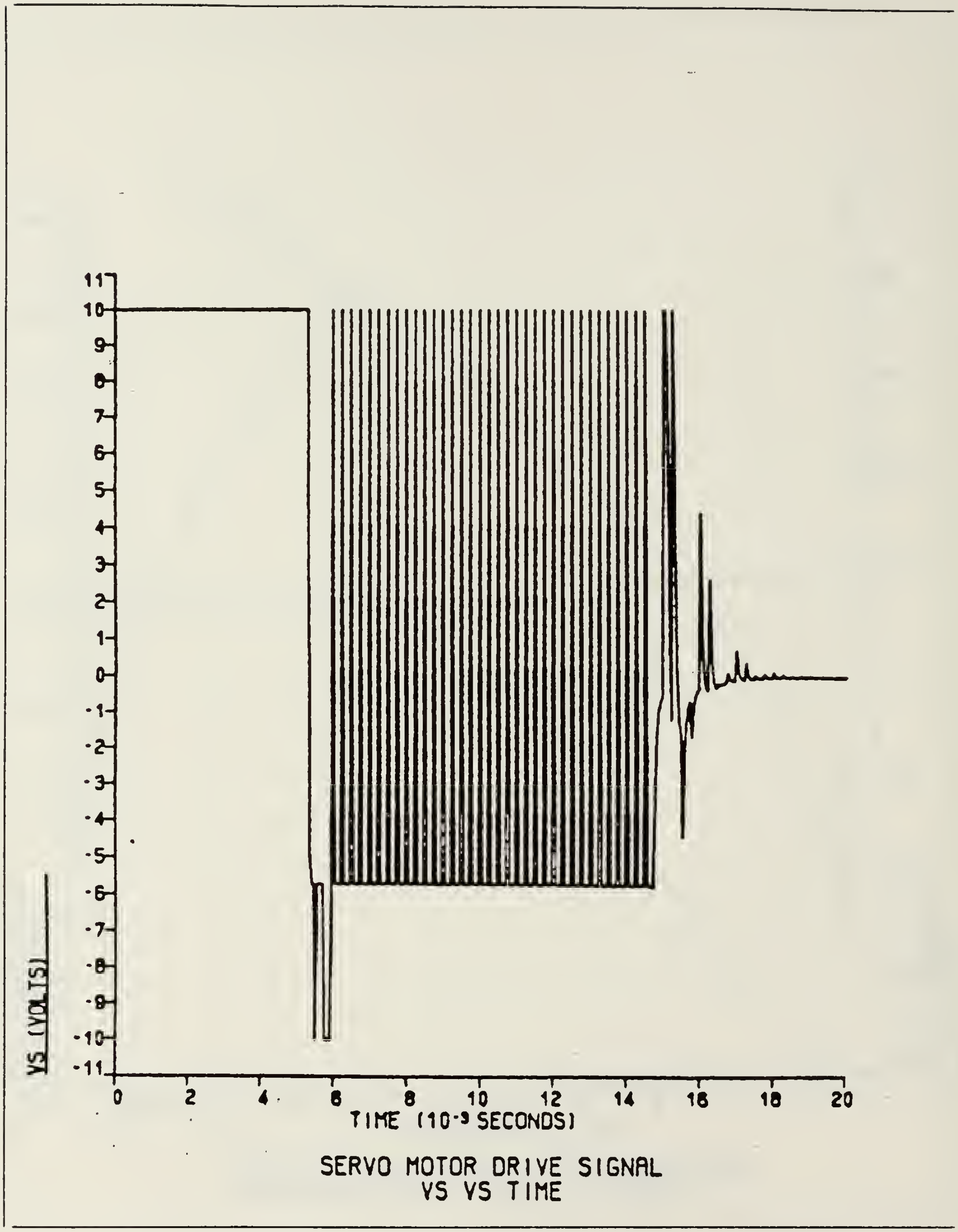

Figure 7.8 Servo Motor Drive Voltage
(100 Track Move)

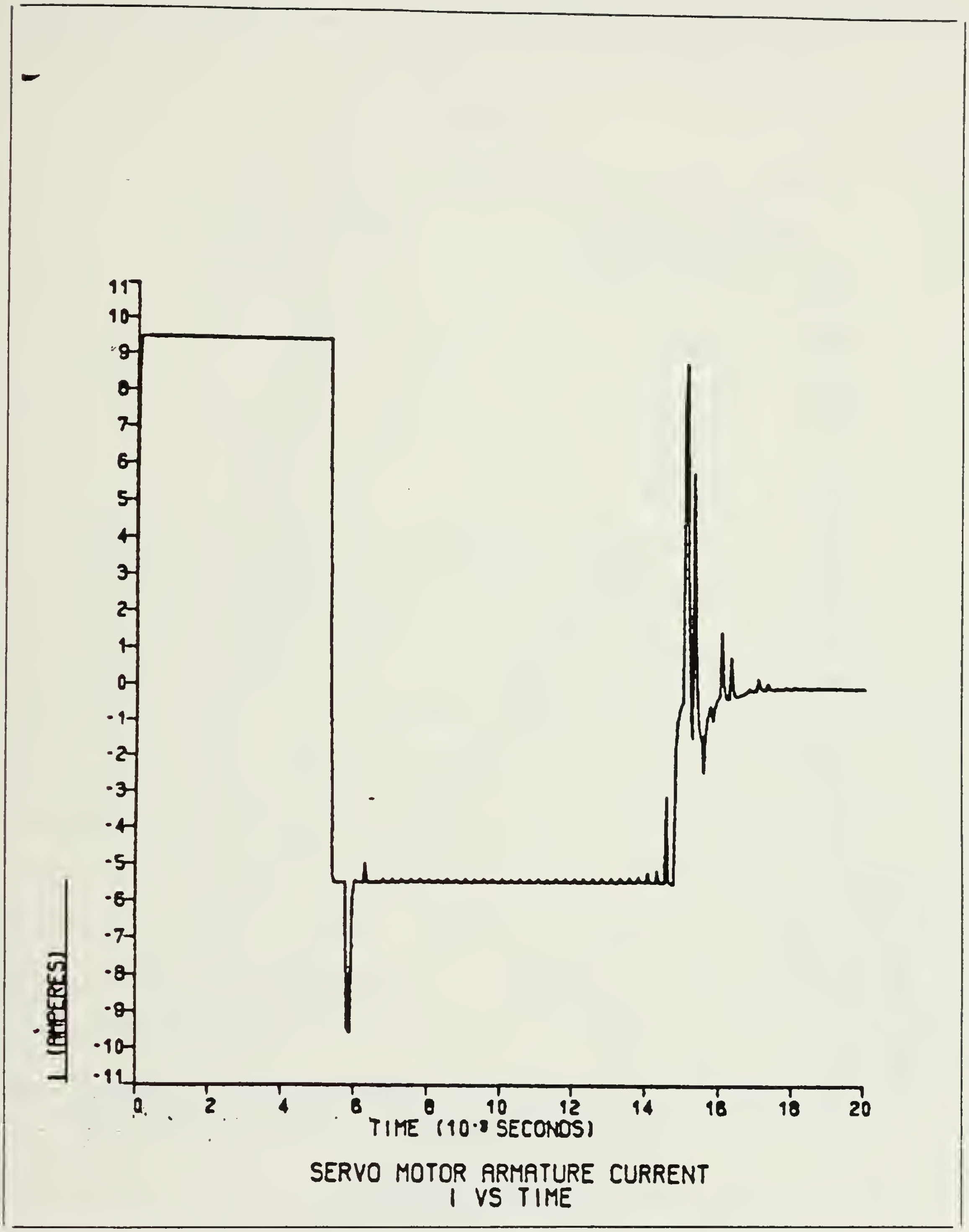

Figure 7.9 Servo Motor Armature Current
(100 Track Move)

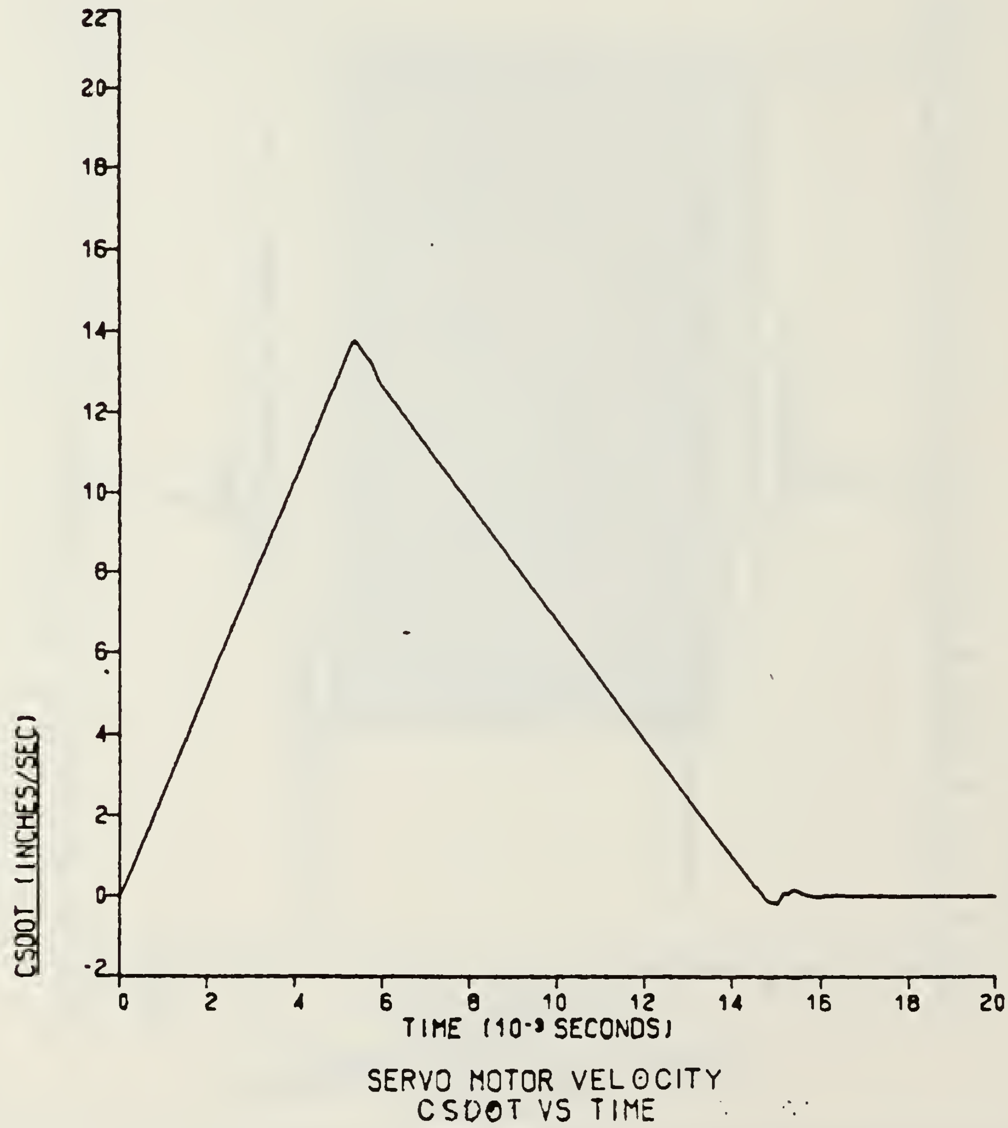

Figure 7.10 Servo Motor Velocity
(100 Track Move)

here. The settling time for moves of 1 to 1000 tracks for the current source drive system are tabulated below.

TABLE 2

Settling Times for Moves of Various Length

Length of Move (tracks)	Settling Times (milliseconds)
1	1.30
5	1.85
10	4.50
50	10.20
100	14.50
500	33.00
1000	46.50

D. EFFECT OF VARYING THE SAMPLING PERIOD

The sampling period (T) has been .25 milliseconds in all preceeding simulations. If the sampling period can be increased and still produce a satisfactory system response two important system factors will be improved. First, increasing the sampling period corresponds to a smaller number of servo information sectors required on the disk surface. With fewer sectors on the disk, more space is available for data storage. Secondly, for each sample of head position, an algorithm is invoked to compute and reset the model states and gain parameter Km. By increasing the sampling period this algorithm is invoked less often for a given move and the microprocessor has fewer computations to perform.

To find a maximum sampling period that insures an acceptable system response, successive simulation trials for

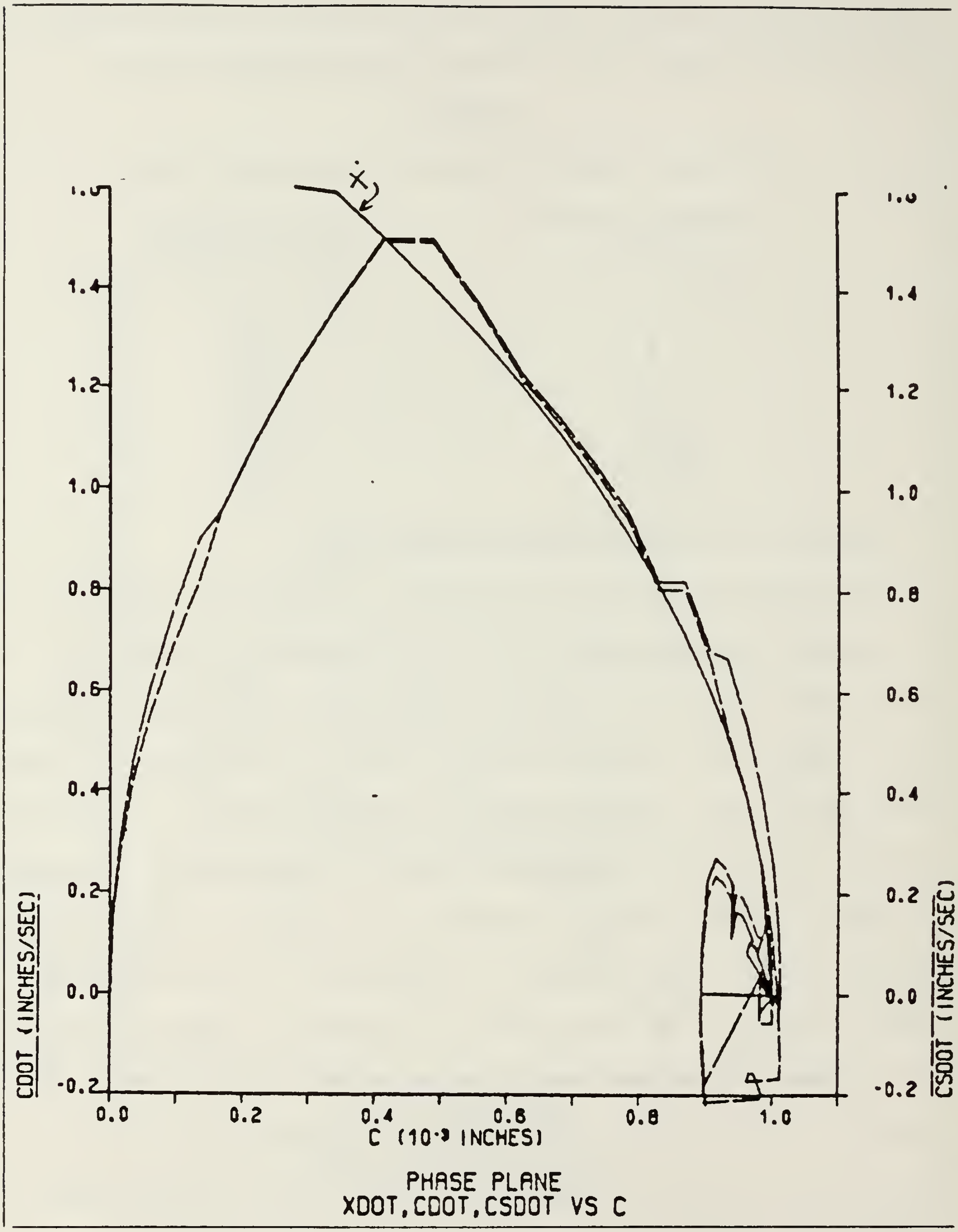

Figure 7.11 Phase Trajectory (T = .35 msec)

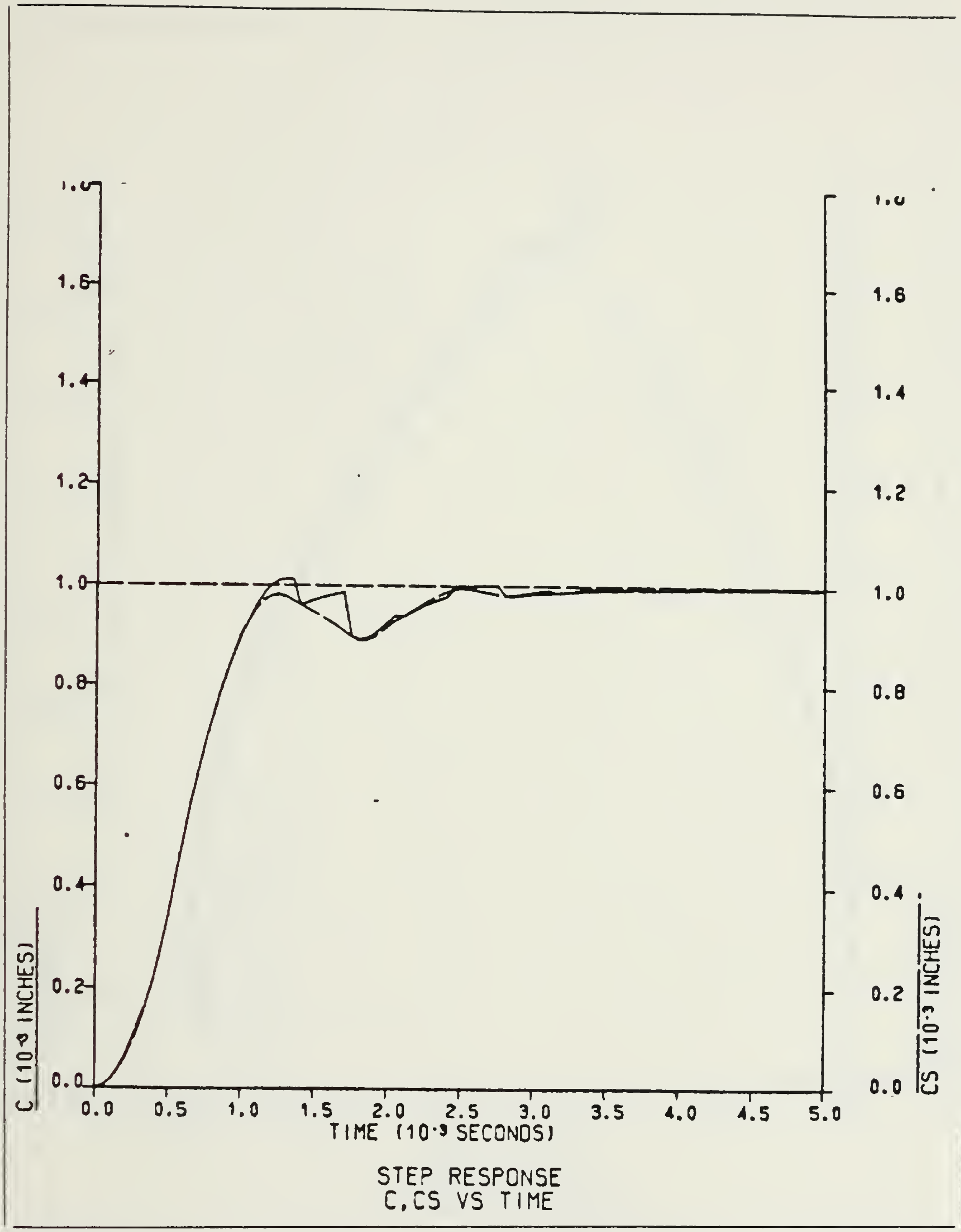

Figure 7.12 Step Response (T = .35 msec)

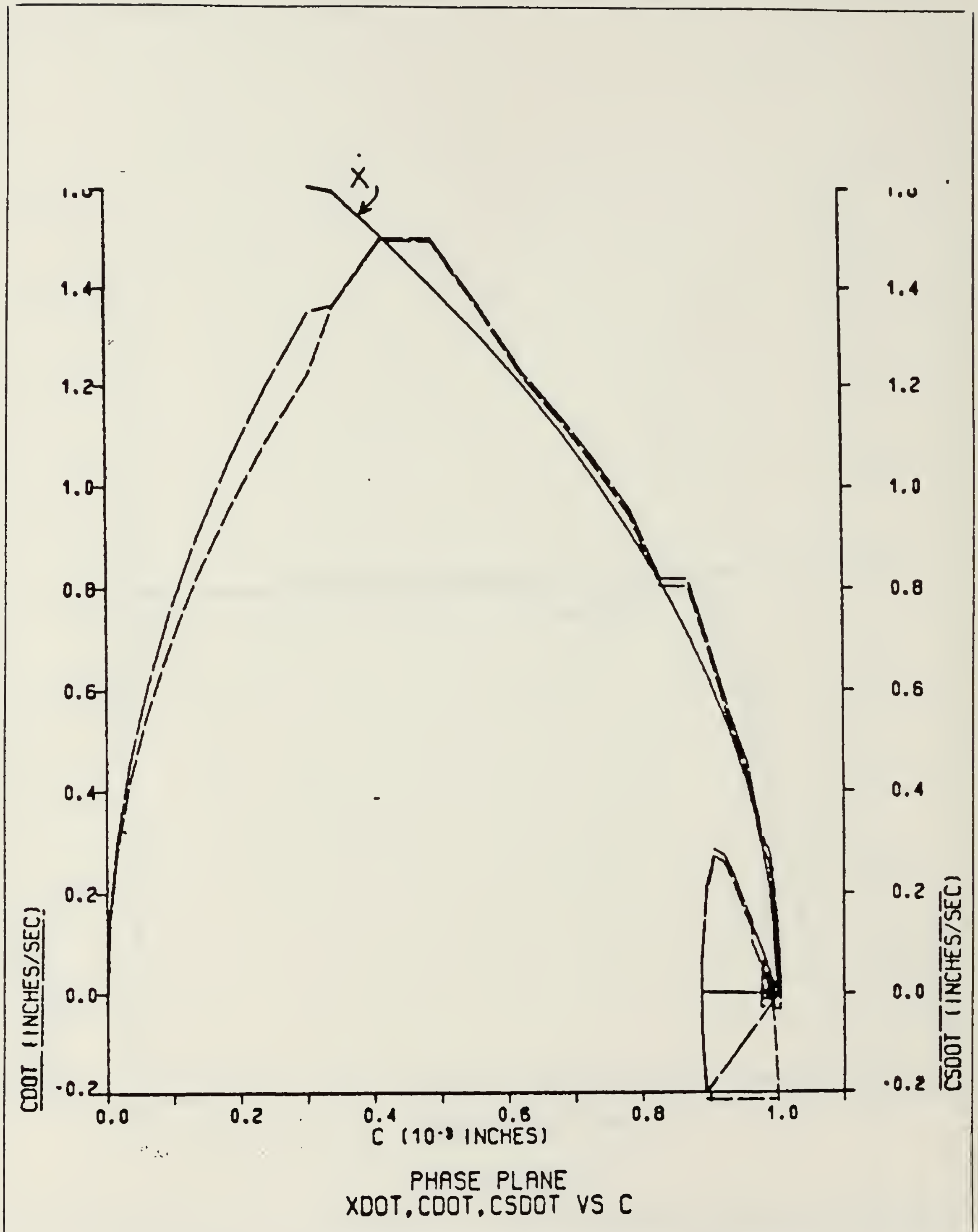

Figure 7.13 Phase Trajectory (T = .50 msec)

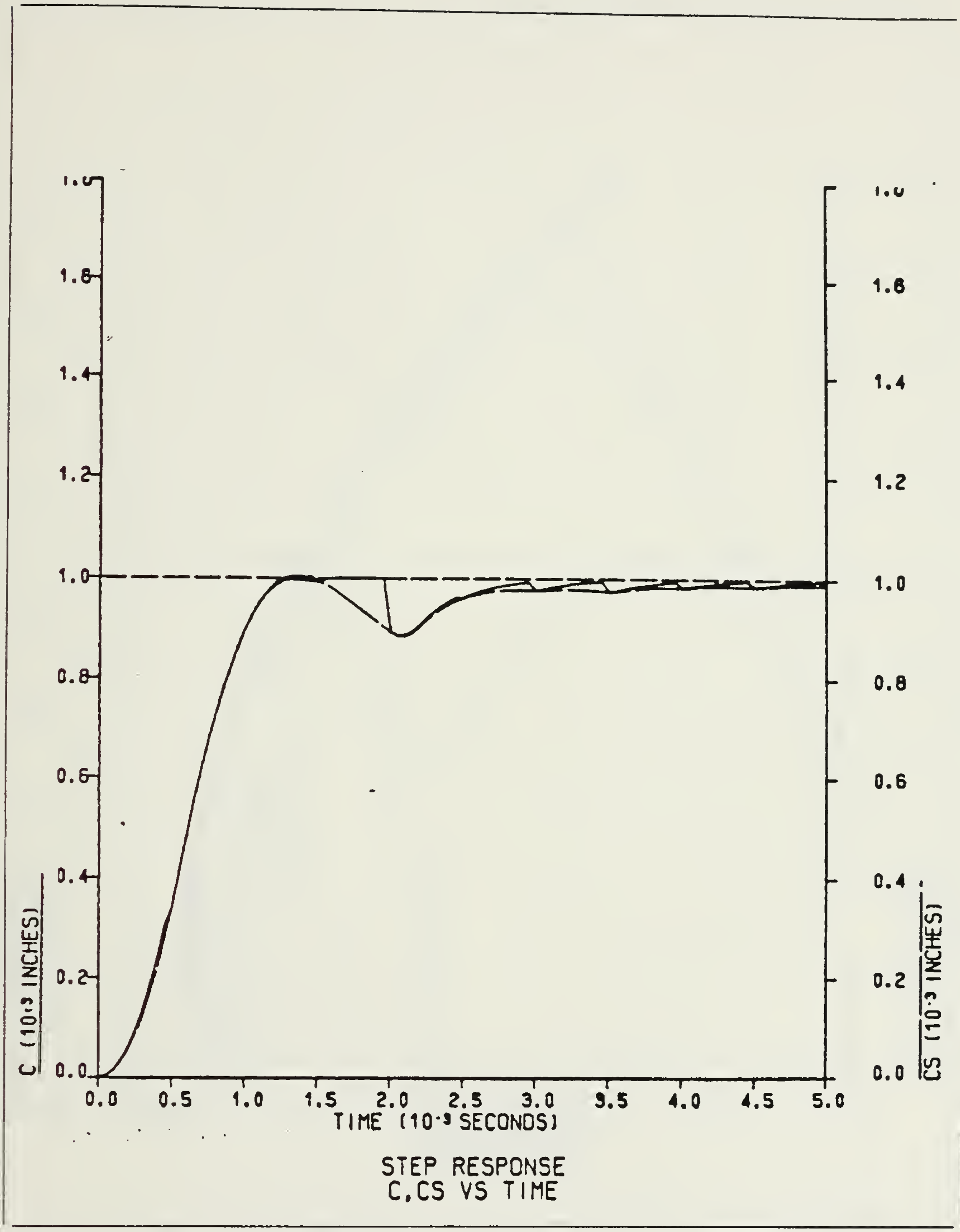

Figure 7.14 Step Response (T = .50 msec)

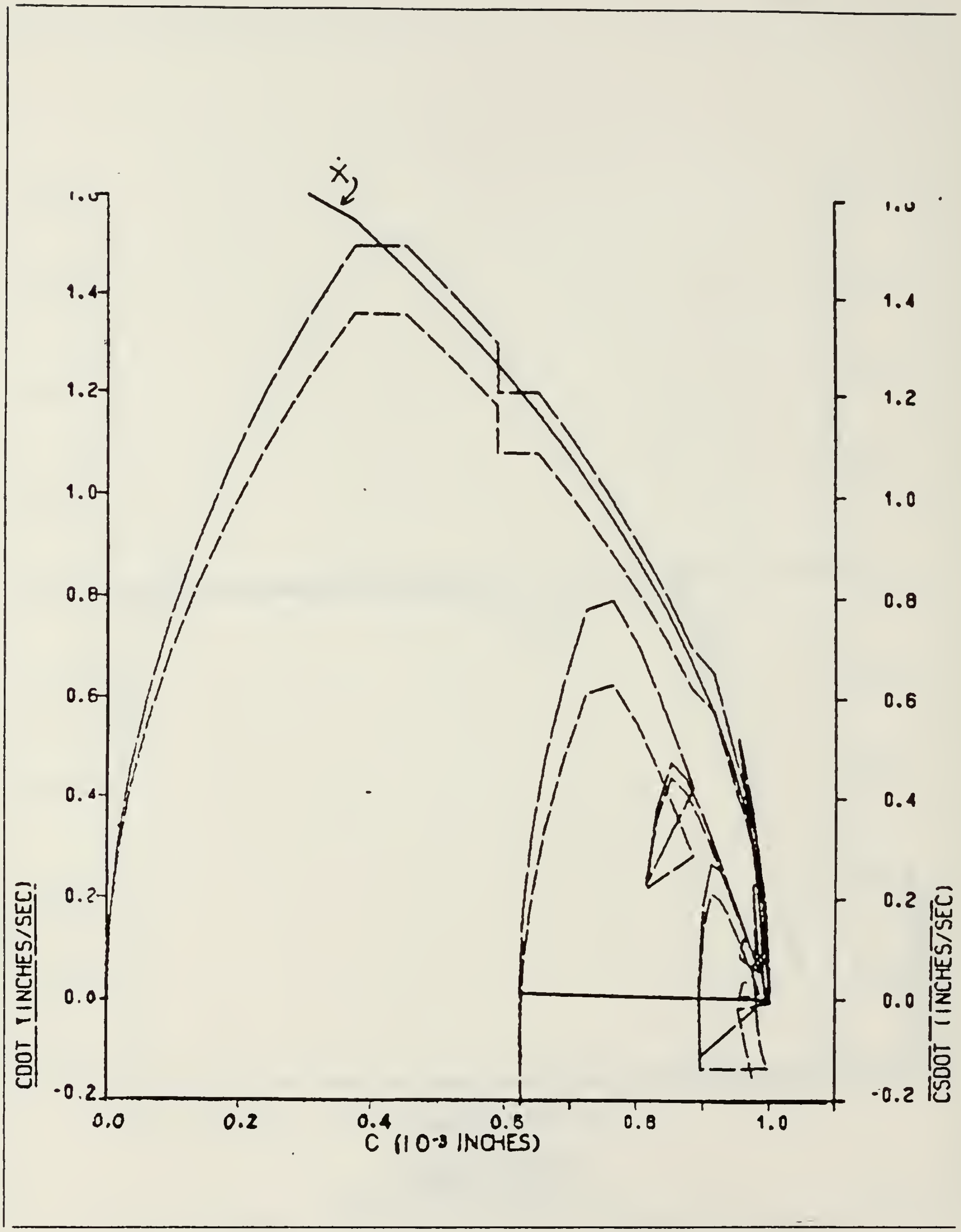

Figure 7.15 Phase Trajectory (T = .70 msec)

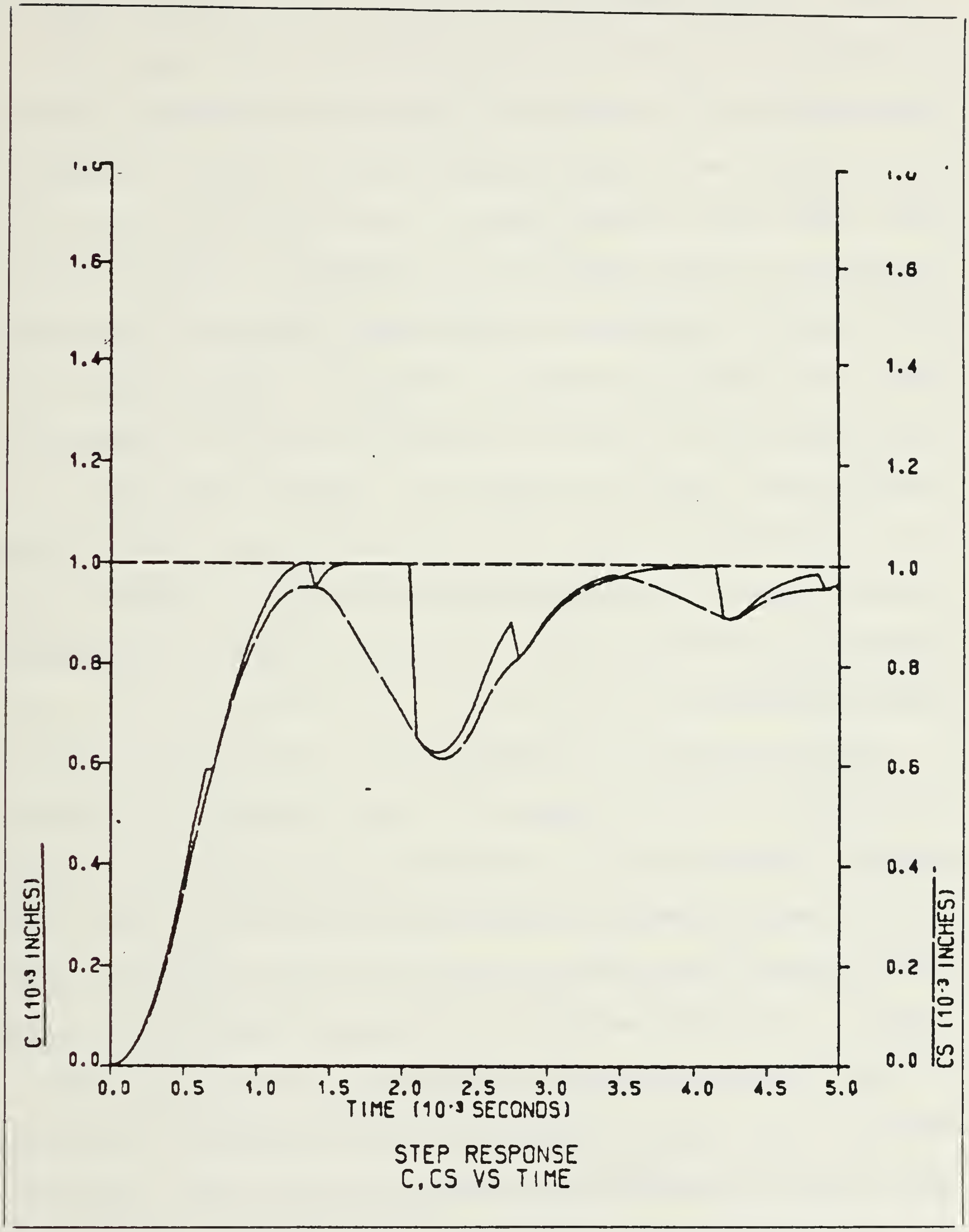

Figure 7.16 Step Response (T = .7 msec)

a one track move were run for T = .35 milliseconds (47 sectors), T = 50 milliseconds (32 sectors), and T = .70 milliseconds (24 sectors). Figures 7.11 through 7.16 show the phase plane plots and step response curves for each of the sampling periods tried. As the figures show, the response of the system is acceptable up to T = .50 milliseconds although for this sampling period the settling time is slightly increased. When the sampling period is increased to T = .70 milliseconds, no position samples are taken during the full acceleration portion of the seek mode and the model and servo motor do not track together during deceleration. The resulting step response shows a large undershoot of about 40%. This would not be an acceptable system response.

By examination of simulation data, the undershoot for all simulation trials appears to be caused by inaccurate updating of servo motor velocity to the model while in track follow. As the sampling period is increased, the servo motor velocity may change signs one or more times between position samples thereby producing an erroneous velocity estimate. If an accurate estimate of servo motor velocity could be made for any sampling period, it may be possible to increase the sampling period beyond .50 milliseconds and still have an acceptable system response. This subject is left as an area for further study.

E. THE EFFECT OF TIME DELAYS ON SYSTEM RESPONSE

The position measurement (CS) has been assumed to be readily available at the sampling instant in the preceeding simulation trials. In an actual system, the head must read the bursts of servo information from the disk and decode them to produce the position measurement. This reading and decoding of position information introduces a time delay into the system. In an actual operating disk file system using servo information sectors similar to that of Figure 1.2, the time delay to read and decode the position information was reported to be approximately 30 microseconds. When this time delay was introduced into the simulation program by use of a DSL/VS TRANSPORT function, only a slight degradation of system performance was noted. With CS delayed by 30 microseconds, the position update to the model is delayed and the velocity and Km updates are derived from this delayed position measurement. Figures 7.17 and 7.18 show the phase plane and step response curves when the position measurement is delayed by 30 microseconds for the one track move (T = .25 milliseconds).

Once the position measurement is made, the model position (C) can be updated but the model velocity ($\dot{C}$) and gain parameter (Km) must be derived by invoking an algorithm. This would introduce an additional time delay into the system due to the calculation time delay. If an arbitrary time delay of 50 microseconds is introduced into the system

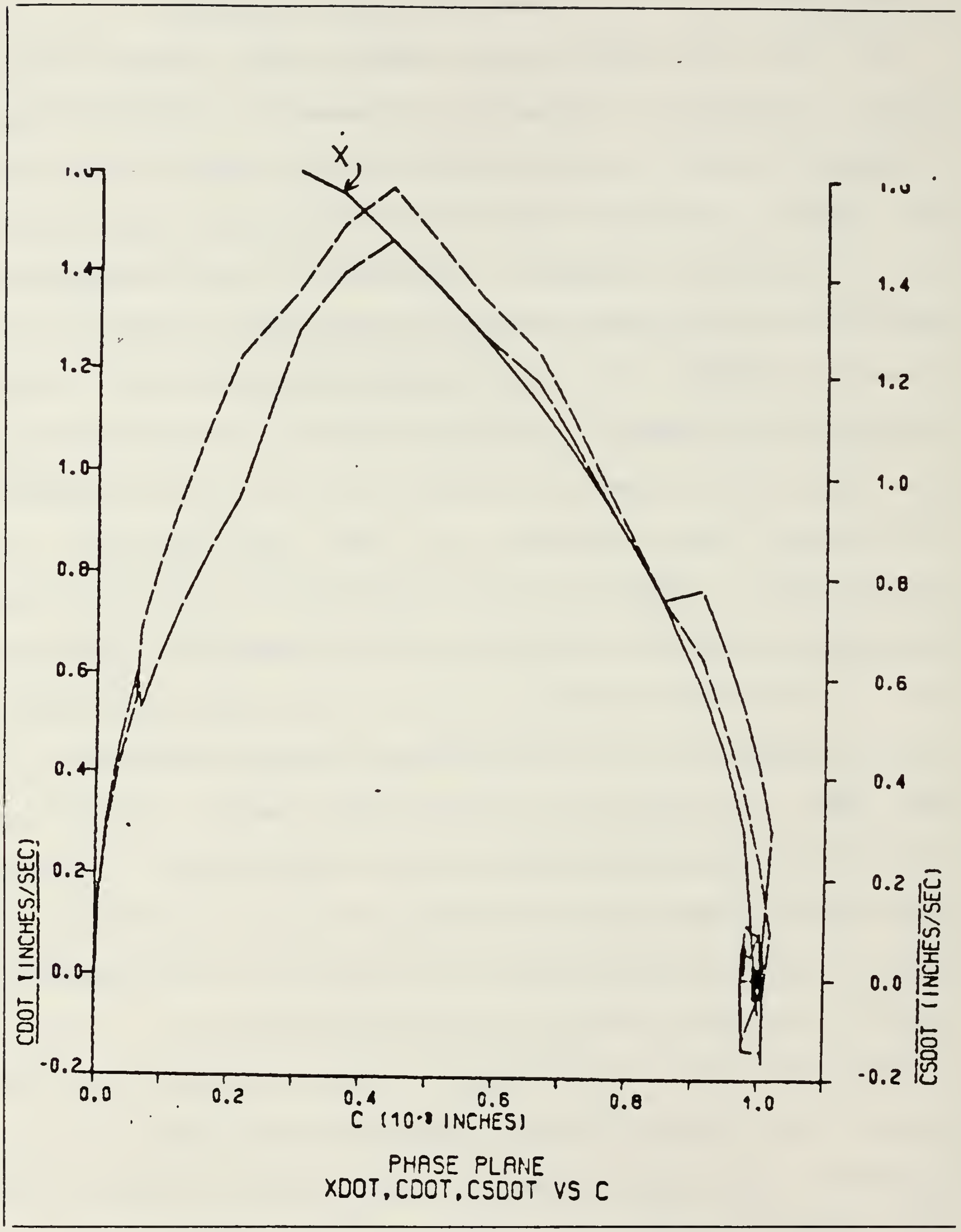

Figure 7.17 Phase Trajectory
(CS Delayed 30 microseconds)

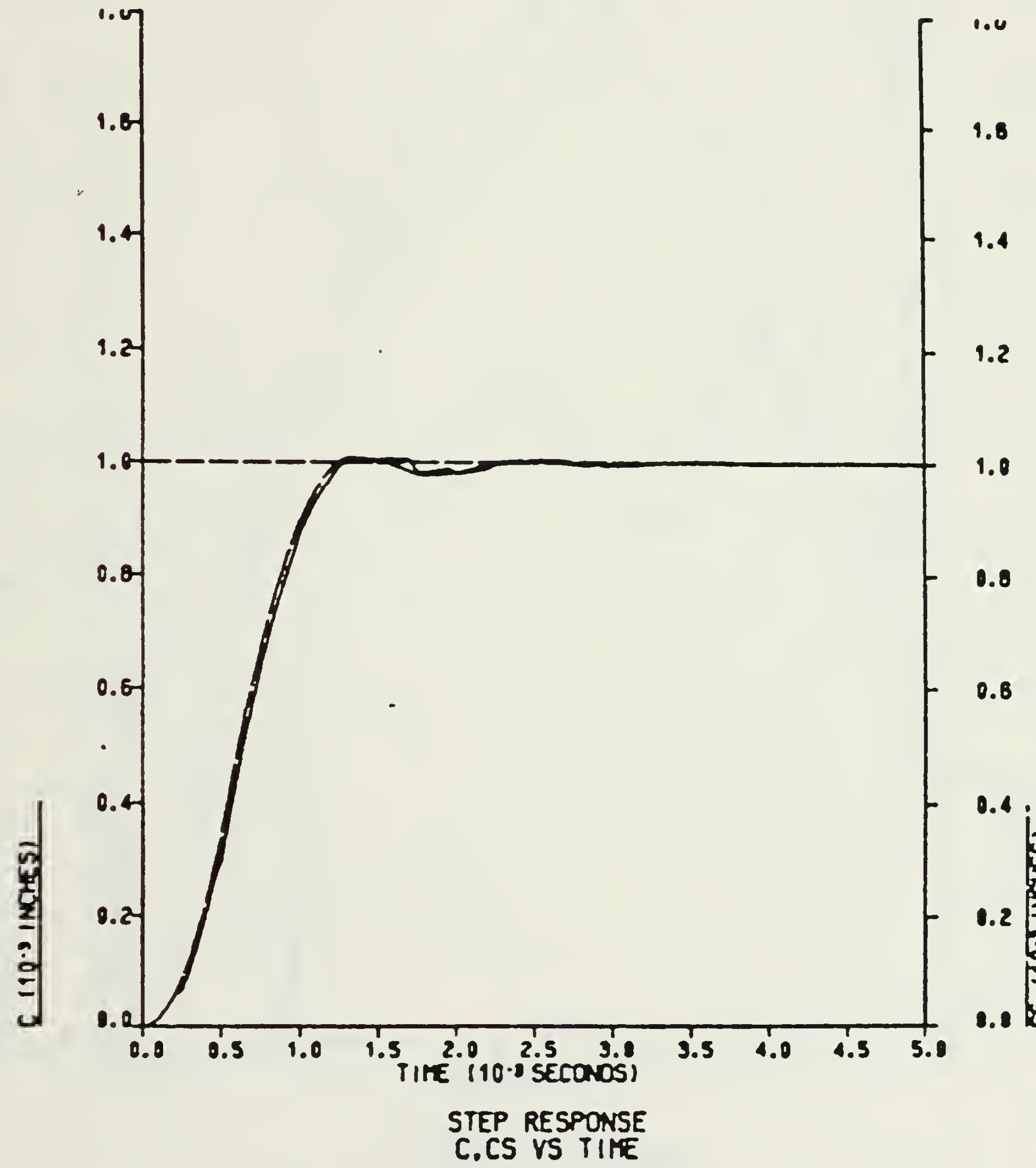

Figure 7.18 Step Response
(CS Delayed 30 microseconds)

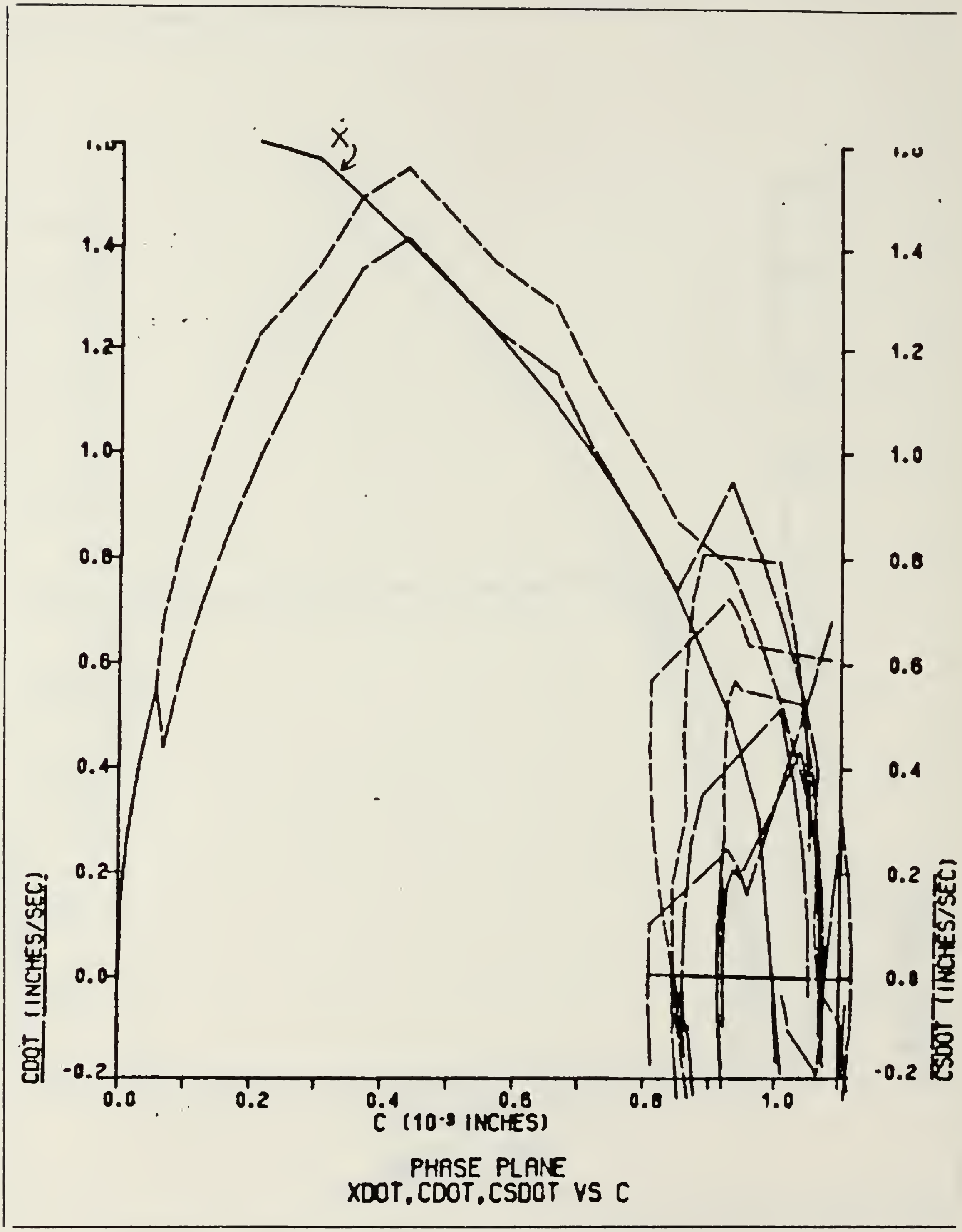

Figure 7.19 Phase Trajectory
(C and Km Updates Delayed
by an Addtional 50 Microseconds)

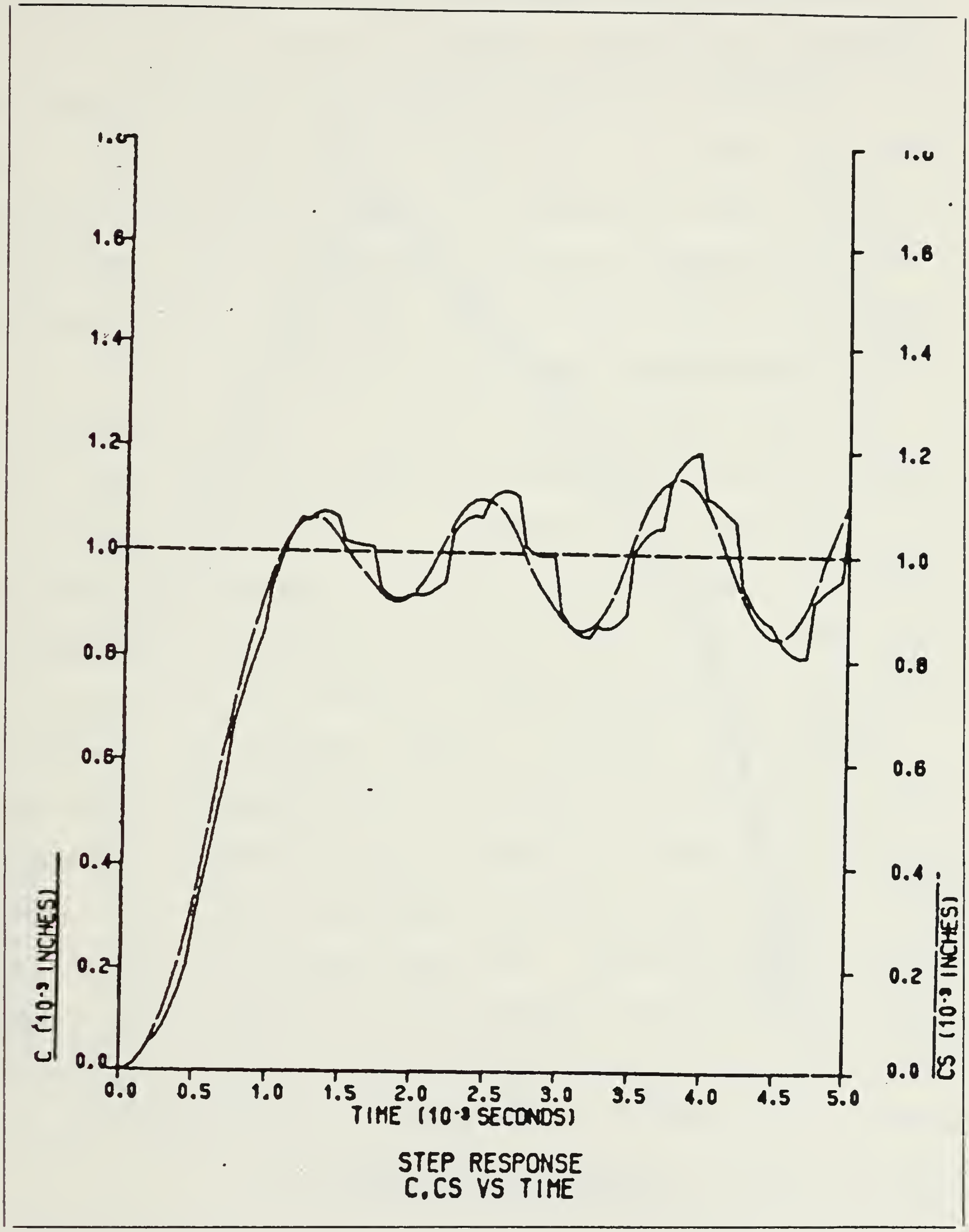

Figure 7.20 Step Response
($\dot{C}$ and Km Updates Delayed
by an Additional 50 Microseconds)

to account for this computation delay before $\dot{C}$ and Km can be updated in the model, stability for the one track move is lost as shown in Figures 7.19 and 7.20. However instability did not occur for the long move of 100 tracks. This indicated that the unstable response for the short move may have been caused by intitial conditions on the compensator (high servo motor velocity over track center) when switching to the track follow mode occurs.

This was shown to be the case. If the value of Km is known from previous moves the model can utilize this value for a short move. In addition, if the model runs only on position updates for the seek mode, the initial conditions upon switching to track follow are reduced and a stable step response is achieved as shown in Figure 7.21 and 7.22.

Reiterating, to obtain a stable response for a short move with an additional time delay of 50 microseconds on model updates of Km and $\dot{C}$, the model is run on updates of position only during the seek mode, and both C and $\dot{C}$ are updated while track following. For long moves ($\geq$ 10 tracks), the model can use all the updates as soon as they are available. Thus, if there is a computation delay involved to update the model, the effects of this time delay can be dealt with to yield a stable response for any length move.

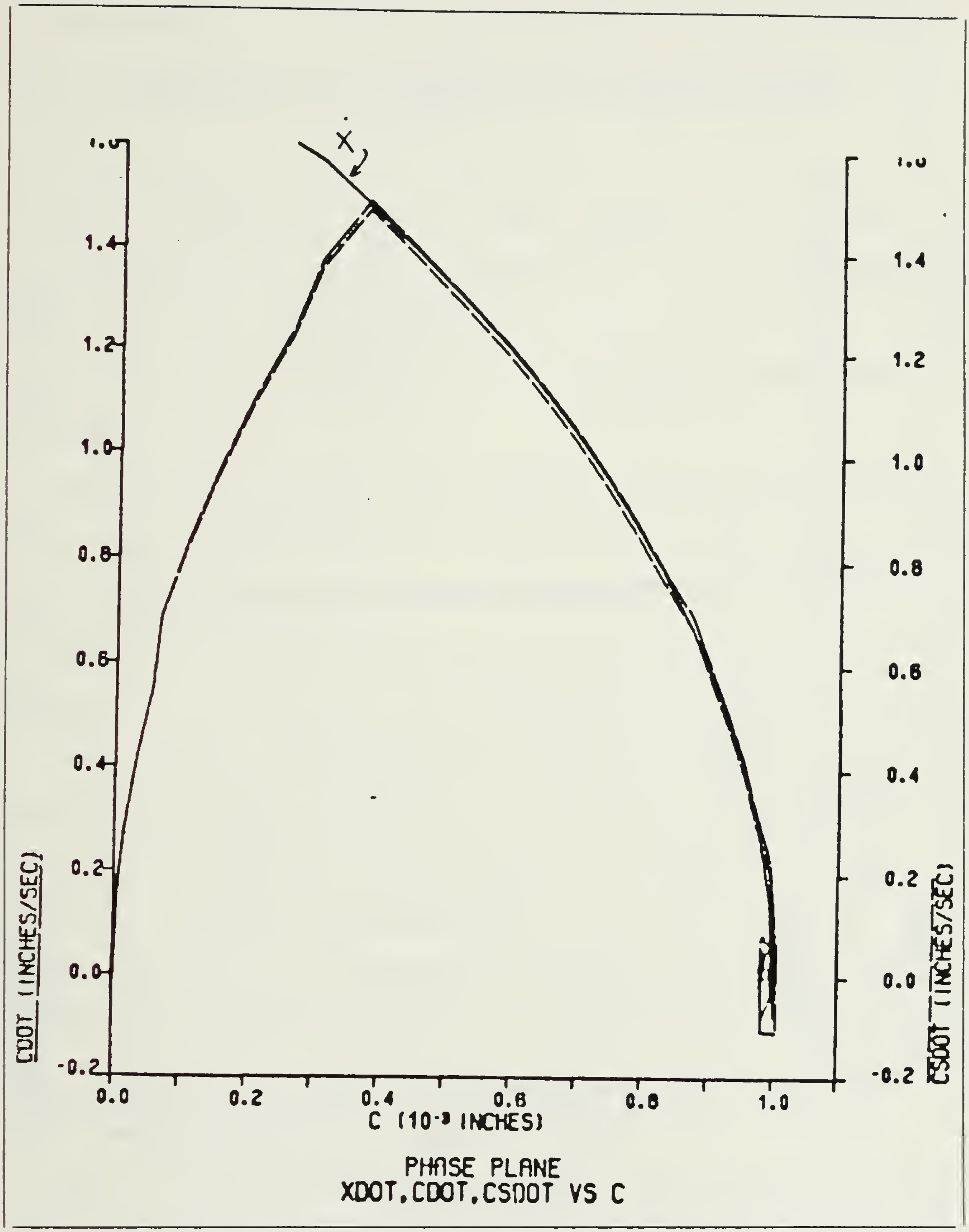

Figure 7.21 Phase Trajectory
(Model Updates of C Only)

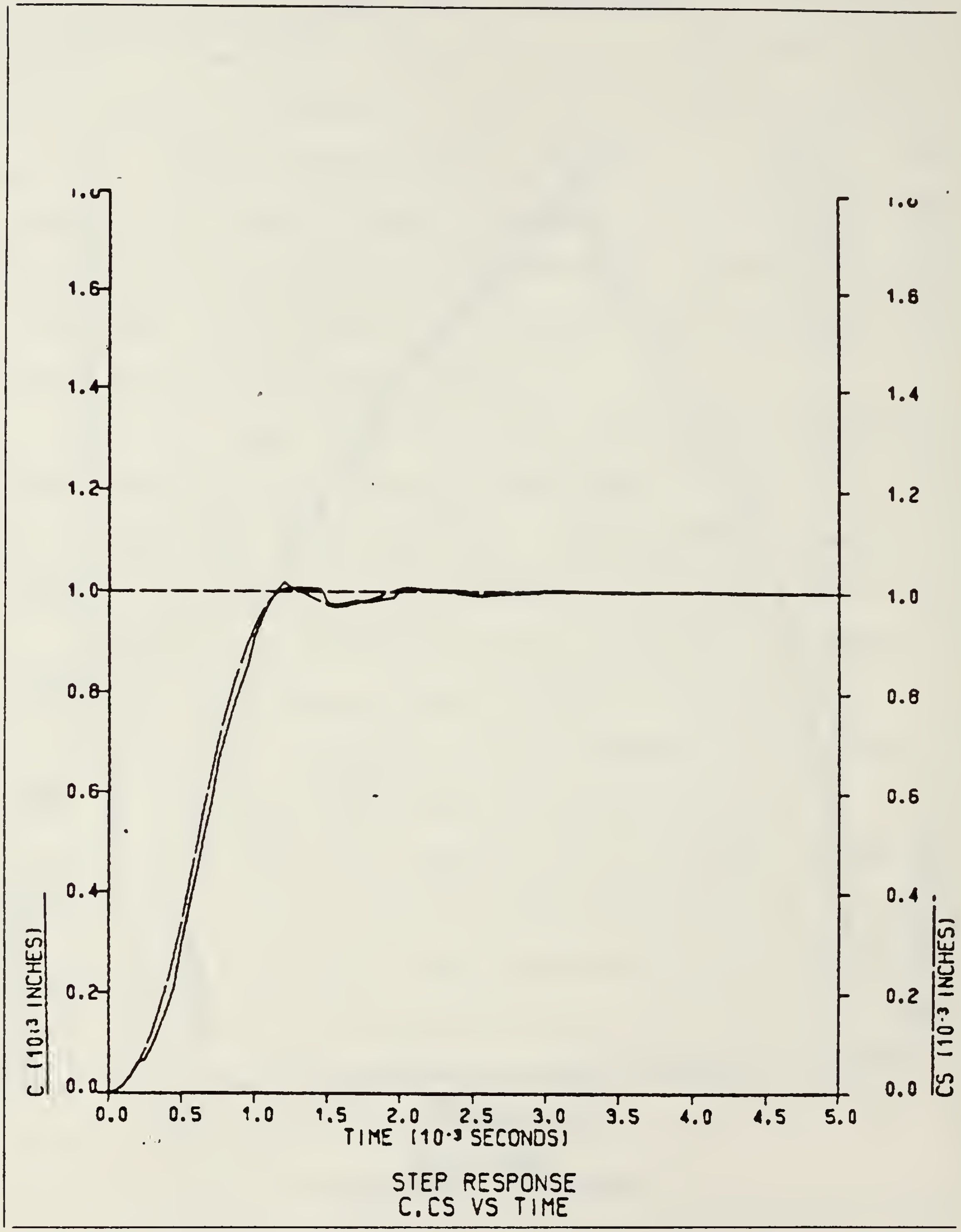

Figure 7.22 Step Response
(Model Updates of C Only)

VIII. IMPLEMENTING THE SYSTEM INTO A MICROPROCESSOR

A. INTRODUCTION

This discussion does not attempt to choose a microprocessor or design a system that will fully implement the adaptive model. Offered here only are a few observations and suggestions that a system designer may find helpful.

The adaptive model must operate in "real-time" because the output of the model is a servo motor drive signal that must be provided at specified time intervals to accurately control the motion of head as it moves across the disk surface. P. Civera, D. Del Corso, and F. Gregoretti [Ref. 4, pp 11-44] have termed this type of system as a "hard real-time" system because "... failure of the computer to complete the required calculations in a specified time will cause a general system failure." For the system under investigation, an inaccurate or delayed servo motor drive signal may cause an overshoot of the desired track or, in the worst possible case, system damage if all control of the servo motor is lost.

DSL/VS simulation of the adaptive model yields a system that is continuous in nature. By using fixed step integration in the simulations with an integration step size that was 1/5 the sampling period, an attempt was made to

simulate how a microprocessor would provide drive signals to the servo motor. At each integration step, the drive signal ($\dot{X}E$) was calculated and fed to the servo motor amplifier. In an actual system this computed drive signal would be made available to the servo amplifier via a D/A converter. If a sufficient number of drive signals are provided during the sampling period, the actions of the servo motor (and head) will be continuous in nature.

B. FLOWCHART OF THE ADAPTIVE MODEL ALGORITHM

One possible method of implementing the adaptive model is to use an infinite loop, where once initiatied, the program flow through the loop continues until interrupted. The cycle time (time to execute the program once through the loop) is critical for the adaptive model. The computations required to decode the bursts of servo information and convert them to a position sample, calculate the head velocity and calculate the gain parameter may disrupt the timing of this loop at the sampling instants. Thus it may be best to use a second microprocessor to do these calculations and provide these values when available by using interrupts to the main microprocessor. The time required to update the values in the main microprocessor by interrupt would be considerably shorter than the main microprocessor performing the decoding and updating.

The flowchart for the adaptive model algorithm is shown in Figure 8.1. This flowchart was based on the DSL/VS

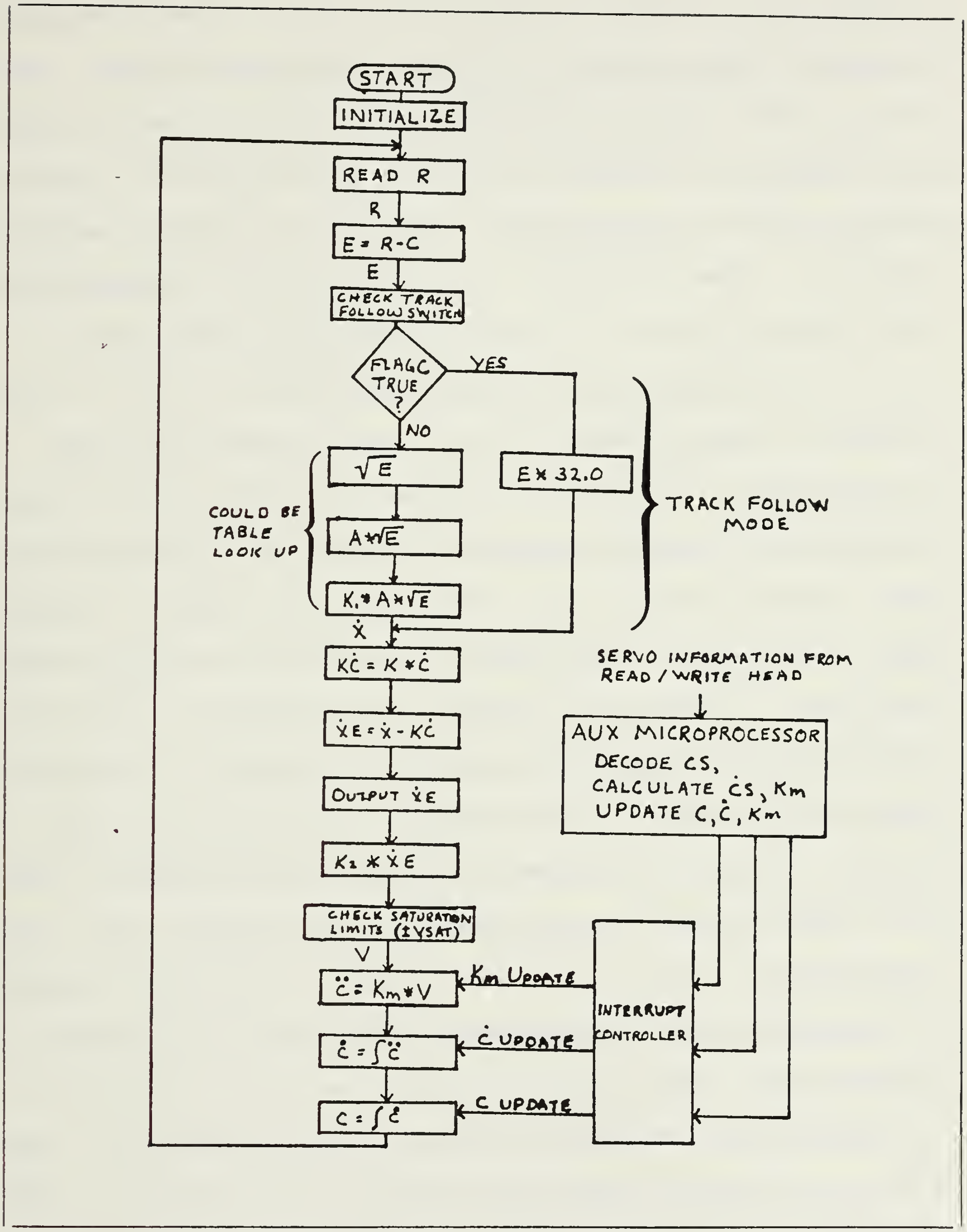

Figure 8.1 Flowchart of the Adaptive Model Algorithm

simulation program of Appendix D where model tachometer feedback compensation is provided during the track follow mode. The auxilliary microprocessor algorithm to compute the updates is not shown in Figure 8.1, but would follow the program flow of the sample region of the afore-mentioned program. When available, the updates of C, $\dot{C}$ and Km are provided by interrupt and the program flow continues.

On system start up, the parameters A, K1, K2, K, FLAG, FLAGC, N, NSW, VSAT, $\dot{C}$, $\dot{X}$ used in the algorithm are initialized (as in the simulation program). Program execution begins when the desired track (R) is read from the main computer CPU. All calculations are made in sequence and the drive signal ($\dot{X}$E) is sent to the servo motor amplifier. the flow through the program continues in a looping fashion providing the drive signal at each cycle pass. When track follow switching conditions are met, the program changes the value of the tachometer feedback gain, (K) and routes around the curve through the gain compensation block. Thus compensation is provided continuously until a different desired track (R) is received from the main computer CPU.

The critical factor in the above discussion is the cycle time to complete one pass through the program. After optimization of the assembled code in the program, one can count the number clock cycles required to complete the calculations in the loop. with this figure in hand

and knowing the number of times the drive signal must be sent to the servo motor amplifier between sampling instants along with the sampling period (determined by the number of sectors on the disk), the clock frequency required for the microprocessor to implement this adaptive model can be determined by

$$f(\text{clock}) = (\#\text{of clock cycles/loop}) \cdot (\#\text{of loops/sampling period})/T \qquad (8.1)$$

Some other factors that are considered when choosing an appropriate microprocessor and associated hardware are the number of I/O ports, the programming language, the amount of built in RAM or ROM on chip, and the word length (word length determines the accuracy of the results). Because numerical integration is used in the model, a built in hardware multiplier may also be a desirable feature. Several microprocessors for control or signal processing applications have D/A and A/D converters on chip. The speed of this conversion will also be an important factor for the system proposed.

While the above discussion offers one possible way to implement the adaptive model, it is by no means the only way. A system designer with considerably more expertise in this area than the author may well find a better way to obtain the desired results. This matter is left as an area for further study.

IX. CONCLUSIONS/AREAS FOR FURTHER STUDY

As a result of the research conducted in this thesis, the controlling of a disk file head positioning servo motor with an adaptive simulation model appears possible. Every effort was made to include possible conditions that may cause the adaptive model system to fail. These included the influence of the electrical time constant on the performance of a voltage drive system, the effect of varying the amplifier parameters, the effect of varying the sampling period and the introduction of time delays into the control system. All these conditions were overcome by offering alternatives to reduce or eliminate the resulting degradation of system response.

One major question still remains. Will the proposed system work when implemented into a microprocessor? This question can only be answered by actually programming a microprocessor with the adaptive model algorithm to control a positioning servo at the speeds and accuracy required for a state of the art disk file system. Continued thesis research in this area is required.

Another area for further study arises from the difficulty in controlling the third order servo motor (with the electrical pole included in the transfer function) during the seek mode with a voltage source drive system. If the

armature circuit of a servo motor can be manufactured so that the inductance has a negligible effect, this problem will be solved. If this inductance cannot be reduced, it may be possible to use a third order servo motor as the model and continue to update only the position and velocity states of the actual servo motor along with an appropriate motor gain adjustment. Adjustment of model poles may not be required if the time constants are initially set to be within the servo motor rated values. Simulation studies of this higher order adaptive model should reveal if this alternative solution is viable.

In the simulation studies, the servo motor saturating amplifier was chosen to be identical to that of the model. Because the outputs of these amplifiers produce the same voltage (V = VS), the model amplifier output can be used to drive the servo motor instead of $\dot{XE}$. This would eliminate the requirement to have an actual amplifier of high gain in the servo motor drive circuit. It would also allow the model to change the gain of the amplifier when in the track following mode when model tachometer feedback compensation is used. If switching conditions are not ideal, limit cycles may occur in the model because of the nonlinear saturating amplifier gain being too high. By having the ability to lower this gain while track following, it may not be necessary to have the head over track center with zero velocity when the switch to track following is made. It may

be possible to switch to track following when the head is 1/10 of a track or more away from track center to reduce the overshoot and eliminate the undershoot noted for the above type of compensation. Simulation studies may show this to be a solution to the problems encountered when the sampling period is increased and time delays are introduced into the system.

APPENDIX A

DSL/VS Program for the Basic Simulation Model of Chapter 2

```
TITLE   APP. A
TITLE   SIMULATION PROGRAM OF THE BASIC MODEL
PARAM   K1=0.80,K2=10000.0,KM=300.00,VSAT=10.,K=1.0
PARAM   REF=0.1000
INITIAL
           PP2=1./20.555
           A=SQRT(2.*KM*VSAT)
           XDOT=0.0
DERIVATIVE
        R=REF*STEP(0.0)
        E=R-C
NOSORT
        IF(E.LT.0.0)XDOT=-A*K1*SQRT(ABS(E))
        IF(E.GE.0.0)XDOT=A*K1*SQRT(E)
SORT
        XDOTE=XDOT-KCDOT
        KCDOT=CDOT*K
        V=LIMIT(-VSAT,VSAT,K2*XDOTE)
        CDDOT=KM*V
        CDOT=INTGRL(0.0,CDDOT)
        C=INTGRL(0.0,CDOT)
TERMINAL
FINISH    C=0.1000
METHOD  RKSFX
CONTRL   FINTIM=0.05000,DELT=0.00005,DELS=.00025
PRINT   0.00005,C,CDOT,V,XDOT
SAVE  (G1)0.00005,TIME,C,XDOT,CDOT
SAVE  (G2)0.0005,TIME,C,REF
GRAPH( G1/G1,DE=TEK618,PO=0.,5) C(LE=8.0,UN='INCHES')....
          XDOT(LE=12,NI=12,LO=0,SC=2.0,AX=OMIT)....
          CDOT(LE=12,NI=12,LO=0,SC=2.0,UN='INCHES/SEC')
GRAPH( G2/G2,DE=TEK618,PO=0.,5) TIME(LE=8.0,NI=5,SC=0.0025....
          UN=SECONDS)....
          C(LE=12,NI=12,LO=0,SC=0.02,UN='INCHES')....
          REF(LE=12,NI=12,LO=0,SC=0.02,AX=OMIT)
*RAPH( G1/G1,DE=TEK618,PO=0.,5) C(LE=8.0,UN='INCHES')....
*         XDOT(LE=12,NI=12,LO=0,SC=.20,AX=OMIT)....
*         CDOT(LE=12,NI=12,LO=0,SC=.20,UN='INCHES/SEC')
*RAPH( G2/G2,DE=TEK618,PO=0.,5) TIME(LE=8.0,NI=5,SC=0.00025....
*         UN=SECONDS)....
*         C(LE=12,NI=12,LO=0,SC=0.0002,UN='INCHES')....
*         REF(LE=12,NI=12,LO=0,SC=0.0002,AX=OMIT)
LABEL  (G1)   PHASE PLANE
LABEL  (G1)   XDOT,CDOT VS C
LABEL  (G2)   STEP RESPONSE
LABEL  (G2)   C VS TIME
END
STOP
```

APPENDIX B

DSL/VS Program for the Adaptive Model of Chapter 3 for Average Head Velocity Updates

DSL/VS Program for the Adaptive Model of Chapter 3 for Head Velocity Updates Using Equation 3.10

DSL/VS Simulation Data for the 100 Track Move

```
TITLE   APP. B(1)
TITLE   SIMULATION PROGRAM OF THE ADAPTIVE MODEL (CSDOT AVG UPDATES)
PARAM   K1=0.80.K2=10000.0.KM=300.00.VSAT=10.000.K=1.0.T=0.00025
PARAM   REF=0.0010
INTGER  N.FLAG
INITIAL
           N=0
           FLAG=0
           PP2=1./20.555
           A=SORT(2.*KM*VSAT)
           XDOT=0.0
DERIVATIVE
         R=REF*STEP(0.0)
         E=R-C
NOSORT
         IF(E.LT.0.0)XDOT=-A*K1*SQRT(ABS(E))
         IF(E.GE.0.0)XDOT=A*K1*SQRT(E)
SORT
         XDOTE=XDOT-KCDOT
         KCDOT=CDOT*K
         V=LIMIT(-VSAT.VSAT.K2*XDOTE)
         CDDOT=KM*V
         CDOT=INTGRL(0.0.CDDOT)
         C=INTGRL(0.0.CDOT)
         VS=LIMIT(-VSAT.VSAT.K2*XDOTE)
         CSDDOT=13.3*VS
         CSDOT=REALPL(0.0.PP2.CSDDOT)
         CS=INTGRL(0.0.CSDOT)
SAMPLE
NOSORT
         IF(N.EQ.0)GO TO 20
         KS=ABS(2.*CS/(VS*((N*T)**2)))
         KM=KS
         C=CS
         CDOT=(CS-CLAST)/T
20       N=N+1
         CLAST=CS
SORT
TERMINAL
FINISH     C=0.0010
METHOD  RKSFX
CONTRL   FINTIM=0.05000.DELT=0.00005.DELS=.00025
PRINT   0.00005.C.CS.CDOT.CSDOT.VS.XDOTE.KM
SAVE  (G1)0.00005.TIME.C.XDOT.CDOT.CSDOT
SAVE  (G2)0.00005.TIME.C.CS.REF
*RAPH(G1/G1.DE=TEK618.PO=0..5) C(LE=7.5.UN='INCHES')....
*         XDOT(LE=12.NI=12.LO=0.SC=2.0.AX=OMIT)....
*         CDOT(LE=12.NI=12.LO=0.SC=2.0.UN='INCHES/SEC')....
```

```
*       CSDOT(LE=12.NI=12.LO=0.PO=7.5.SC=2.0.UN='INCHES/SEC')
*GRAPH( G2/G2.DE=TEK618.PO=0..5) TIME(LE=7.5.NI=6.SC=0.0025....
*       UN=SECONDS)....
*       C(LE=12.NI=12.LO=0.SC=0.02.UN='INCHES')....
*       CS(LE=12.NI=12.LO=0.PO=7.5.SC=0.02.UN='INCHES')....
*       REF(LE=12.NI=12.LO=0.SC=0.02.AX=OMIT)
GRAPH( G1/G1.DE=TEK618.PO=0..5) C(LE=7.0.SC=0.0001.NI=11.UN='INCHES')....
        XDOT(LE=12.NI=12.LO=0.SC=.20.AX=OMIT)....
        CDOT(LE=12.NI=12.LO=0.SC=.20.UN='INCHES/SEC')....
        CSDOT(LE=12.NI=12.LO=0.PO=7.0.SC=.20.UN='INCHES/SEC')
GRAPH( G2/G2.DE=TEK618.PO=0..5) TIME(LE=7.0.NI=11.SC=0.0001....
        UN=SECONDS)....
        C(LE=12.NI=12.LO=0.SC=0.0002.UN='INCHES')....
        CS(LE=12.NI=12.LO=0.PO=7.0.SC=0.0002.UN='INCHES')....
        REF(LE=12.NI=12.LO=0.SC=0.0002.AX=OMIT)
LABEL (G1)  PHASE PLANE
LABEL (G1)  XDOT.CDOT.CSDOT VS C
LABEL (G2)  STEP RESPONSE
LABEL (G2)  C.CS VS TIME
END
STOP
```

```
TITLE   APP. B(2)
TITLE   SIMULATION PROGRAM OF THE ADAPTIVE MODEL (CSDOT EQN.3.10 UPDATES)
PARAM   K1=0.80.K2=10000.0.KM=300.00.VSAT=10.00.K=1.0.T=0.00025
PARAM   REF=0.0010
INTGER  N.FLAG
INITIAL
          N=0
          FLAG=0
          PP2=1./20.555
          A=SQRT(2.*KM*VSAT)
          XDOT=0.0
          CDOT=0.0
DERIVATIVE
        R=REF*STEP(0.0)
        E=R-C
NOSORT
        IF(E.LT.0.0)XDOT=-A*K1*SQRT(ABS(E))
        IF(E.GE.0.0)XDOT=A*K1*SQRT(E)
SORT
        XDOTE=XDOT-KCDOT
        KCDOT=CDOT*K
        V=LIMIT(-VSAT.VSAT.K2*XDOTE)
NOSORT
        IF(FLAG.EQ.1)GO TO 5
        IF(V.LT.VSAT.AND.TIME.GT.0.00005)FLAG=1
        NSW=N
5       CONTINUE
SORT
        CDDOT=KM*V
        CDOT=INTGRL(0.0.CDDOT)
        C=INTGRL(0.0.CDOT)
        VS=LIMIT(-VSAT.VSAT.K2*XDOTE)
        CSDDOT=13.3*VS
        CSDOT=REALPL(0.0.PP2.CSDDOT)
        CS=INTGRL(0.0.CSDOT)
SAMPLE
NOSORT
        IF(N.EQ.0)GO TO 20
        C=CS
        KS=ABS(2.*CS/(VS*((N*T)**2)))
        IF(FLAG.EQ.0)KM=KS
        IF(N.GE.2)CSDOTL=(CS-CS2L)/(2.*T)
        IF (FLAG.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
        IF(N.EQ.NSW.AND.FLAG.EQ.1)GO TO 20
        IF (FLAG.EQ.1)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
20      N=N+1
        CSDOTL=CDOT
        CS2L=CSLAST
```

```
      CSLAST=CS
SORT
TERMINAL
FINISH    C=0.0010
METHOD RKSFX
CONTRL   FINTIM=0.05000,DELT=0.00005,DELS=.00025
PRINT   0.00005,C,CS,CDOT,CSDOT,CSDOTL,CSLAST,NSW,N,VS,XDOTE,KM
SAVE (G1)0.00005,TIME,C,XDOT,CDOT,CSDOT
SAVE (G3)0.00005,TIME,C,CS,REF
*RAPH(G1/G1,DE=TEK618,PO=0,.5) C(LE=7.5,UN='INCHES')....
*          XDOT(LE=12,NI=12,LO=-2.,SC=2.0,AX=OMIT)....
*          CDOT(LE=12,NI=12,LO=-2.0,SC=2.0,UN='INCHES/SEC')....
*          CSDOT(LE=12,NI=12,LO=-2.0,PO=7.5,SC=2.0,UN='INCHES/SEC')
*RAPH(G3/G3,DE=TEK618,PO=0,.5) TIME(LE=7.5,NI=6,SC=0.0025,...
*          UN=SECONDS)....
*          C(LE=12,NI=12,LO=0,SC=0.02,UN='INCHES')....
*          CS(LE=12,NI=12,LO=0,PO=7.5,SC=0.02,UN='INCHES')....
*          REF(LE=12,NI=12,LO=0,SC=0.02,AX=OMIT)
GRAPH(G1/G1,DE=TEK618,PO=0,.5) C(LE=7.0,SC=0.0001,NI=11,UN='INCHES')....
           XDOT(LE=12,NI=12,LO=-.20,SC=.20,AX=OMIT)....
           CDOT(LE=12,NI=12,LO=-.20,SC=.20,UN='INCHES/SEC')....
           CSDOT(LE=12,NI=12,LO=-.20,PO=7.0,SC=.20,UN='INCHES/SEC')
GRAPH(G3/G3,DE=TEK618,PO=0,.5) TIME(LE=7.0,NI=13,SC=0.0001....
           UN=SECONDS)....
           C(LE=12,NI=12,LO=0,SC=0.0002,UN='INCHES')....
           CS(LE=12,NI=12,LO=0,PO=7.0,SC=0.0002,UN='INCHES')....
           REF(LE=12,NI=12,LO=0,SC=0.0002,AX=OMIT)
LABEL (G1)   PHASE PLANE
LABEL (G1)   XDOT,CDOT,CSDOT  VS C
LABEL (G3)   STEP RESPONSE
LABEL (G3)   C,CS  VS TIME
END
STOP
```

```
              >>>> DSL SIMULATION INPUT DATA <<<<
TITLE   APP. B(3)
TITLE     DATA FOR THE ADAPTIVE MODEL (CSDOT EQN.3.10 UPDATES)
PARAM   K1=0.80,K2=10000.0,KM=300.00,VSAT=10.00,K=1.0,T=0.0002
PARAM   REF=0.1000
INTGER N,FLAG
FINISH    C=0.1000
METHOD RKSFX
CONTRL   FINTIM=0.05000,DELT=0.00005,DELS=.00025
PRINT   0.00005,C,CS,CDOT,CSDOT,KM
SAVE (G1)0.00005,TIME,C,XDOT,CDOT,CSDOT
SAVE (G3)0.00005,TIME,C,CS,REF
GRAPH(G1/G1,DE=TEK618,PO=0.,5) C(LE=7.5,UN='INCHES')....
        XDOT(LE=12,NI=12,LO=-2.,SC=2.0,AX=OMIT)....
        CDOT(LE=12,NI=12,LO=-2.0,SC=2.0,UN='INCHES/SEC')....
        CSDOT(LE=12,NI=12,LO=-2.0,PO=7.5,SC=2.0,UN='INCHES/SEC')
GRAPH(G3/G3,DE=TEK618,PO=0.,5) TIME(LE=7.5,NI=6,SC=0.0025....
        UN=SECONDS)....
        C(LE=12,NI=12,LO=0,SC=0.02,UN='INCHES')....
        CS(LE=12,NI=12,LO=0,PO=7.5,SC=0.02,UN='INCHES')....
        REF(LE=12,NI=12,LO=0,SC=0.02,AX=OMIT)
LABEL (G1)   PHASE PLANE
LABEL (G1)   XDOT,CDOT,CSDOT VS C
LABEL (G3)   STEP RESPONSE
LABEL (G3)   C,CS VS TIME
END
****  RKSFX    INTEGRATION METHOD USED ***
      *** DSL OUTPUT LISTING, GROUP  1
       APP. B(2)
       SIMULATION PROGRAM OF THE ADAPTIVE MODEL (CSDOT EQN.3.10 UPDATES)
```

TIME	C	CS	CDOT	CSDOT	KM
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	300.00
5.00000E-05	3.75000E-06	3.41610E-06	0.15000	0.13662	300.00
1.00000E-04	1.50000E-05	1.36597E-05	0.30000	0.27310	300.00
1.50000E-04	3.37500E-05	3.07238E-05	0.45000	0.40944	300.00
2.00000E-04	6.00000E-05	5.46015E-05	0.60000	0.54564	300.00
2.50000E-04	8.52856E-05	8.52856E-05	0.68228	0.68170	272.91
3.00000E-04	1.22811E-04	1.22769E-04	0.81874	0.81762	272.91
3.50000E-04	1.67160E-04	1.67045E-04	0.95520	0.95340	272.91
4.00000E-04	2.18331E-04	2.18107E-04	1.0917	1.0890	272.91
4.50000E-04	2.76325E-04	2.75947E-04	1.2281	1.2245	272.91
5.00000E-04	3.40559E-04	3.40559E-04	1.3611	1.3599	272.45
5.50000E-04	4.12018E-04	4.11936E-04	1.4973	1.4951	272.45
6.00000E-04	4.90289E-04	4.90070E-04	1.6335	1.6302	272.45
6.50000E-04	5.75370E-04	5.74955E-04	1.7697	1.7652	272.45
7.00000E-04	6.67263E-04	6.66584E-04	1.9060	1.9000	272.45
7.50000E-04	7.64950E-04	7.64950E-04	2.0358	2.0346	271.98
8.00000E-04	8.70139E-04	8.70045E-04	2.1718	2.1692	271.98

8.50000E-04	9.82128E-04	9.81864E-04	2.3078	2.3036	271.98
9.00000E-04	1.10092E-03	1.10040E-03	2.4438	2.4378	271.98
9.50000E-04	1.22650E-03	1.22564E-03	2.5798	2.5719	271.98
1.00000E-03	1.35759E-03	1.35759E-03	2.7071	2.7059	271.52
1.05000E-03	1.49634E-03	1.49623E-03	2.8428	2.8398	271.52
1.10000E-03	1.64187E-03	1.64156E-03	2.9786	2.9735	271.52
1.15000E-03	1.79419E-03	1.79358E-03	3.1143	3.1070	271.52
1.20000E-03	1.95331E-03	1.95226E-03	3.2501	3.2404	271.52
1.25000E-03	2.11762E-03	2.11762E-03	3.3749	3.3737	271.06
1.30000E-03	2.28975E-03	2.28963E-03	3.5104	3.5069	271.06
1.35000E-03	2.46866E-03	2.46831E-03	3.6459	3.6399	271.06
1.40000E-03	2.65434E-03	2.65362E-03	3.7815	3.7728	271.06
1.45000E-03	2.84681E-03	2.84558E-03	3.9170	3.9055	271.06
1.50000E-03	3.04418E-03	3.04418E-03	4.0393	4.0381	270.59
1.55000E-03	3.24952E-03	3.24939E-03	4.1746	4.1706	270.59
1.60000E-03	3.46163E-03	3.46124E-03	4.3099	4.3030	270.59
1.65000E-03	3.68051E-03	3.67969E-03	4.4452	4.4352	270.59
1.70000E-03	3.90615E-03	3.90475E-03	4.5805	4.5672	270.59
1.75000E-03	4.13641E-03	4.13641E-03	4.7003	4.6992	270.13
1.80000E-03	4.37480E-03	4.37466E-03	4.8354	4.8309	270.13
1.85000E-03	4.61994E-03	4.61950E-03	4.9704	4.9626	270.13
1.90000E-03	4.87184E-03	4.87092E-03	5.1055	5.0941	270.13
1.95000E-03	5.13049E-03	5.12891E-03	5.2406	5.2255	270.13
2.00000E-03	5.39347E-03	5.39347E-03	5.3579	5.3568	269.67
2.05000E-03	5.66473E-03	5.66459E-03	5.4927	5.4879	269.67
2.10000E-03	5.94274E-03	5.94226E-03	5.6276	5.6189	269.67
2.15000E-03	6.22749E-03	6.22647E-03	5.7624	5.7497	269.67
2.20000E-03	6.51898E-03	6.51722E-03	5.8972	5.8804	269.67
2.25000E-03	6.81451E-03	6.81451E-03	6.0121	6.0110	269.22
2.30000E-03	7.11848E-03	7.11832E-03	6.1467	6.1415	269.22
2.35000E-03	7.42919E-03	7.42865E-03	6.2813	6.2718	269.22
2.40000E-03	7.74662E-03	7.74550E-03	6.4160	6.4019	269.22
2.45000E-03	8.07078E-03	8.06885E-03	6.5506	6.5320	269.22
2.50000E-03	8.39870E-03	8.39870E-03	6.6630	6.6619	268.76
2.55000E-03	8.73521E-03	8.73504E-03	6.7974	6.7917	268.76
2.60000E-03	9.07844E-03	9.07786E-03	6.9318	6.9213	268.76
2.65000E-03	9.42838E-03	9.42717E-03	7.0662	7.0508	268.76
2.70000E-03	9.78505E-03	9.78294E-03	7.2005	7.1802	268.76
2.75000E-03	1.01452E-02	1.01452E-02	7.3106	7.3095	268.30
2.80000E-03	1.05141E-02	1.05139E-02	7.4447	7.4386	268.30
2.85000E-03	1.08897E-02	1.08890E-02	7.5789	7.5675	268.30
2.90000E-03	1.12720E-02	1.12706E-02	7.7130	7.6964	268.30
2.95000E-03	1.16610E-02	1.16587E-02	7.8472	7.8251	268.30
3.00000E-03	1.20531E-02	1.20531E-02	7.9548	7.9537	267.85
3.05000E-03	1.24542E-02	1.24540E-02	8.0887	8.0821	267.85
3.10000E-03	1.28620E-02	1.28614E-02	8.2226	8.2105	267.85
3.15000E-03	1.32765E-02	1.32751E-02	8.3566	8.3386	267.85
3.20000E-03	1.36977E-02	1.36952E-02	8.4905	8.4667	267.85

3.25000E-03	1.41218E-02	1.41218E-02	8.5957	8.5946	257.39
3.30000E-03	1.45549E-02	1.45547E-02	8.7294	8.7224	257.39
3.35000E-03	1.49947E-02	1.49940E-02	8.8631	8.8501	257.39
3.40000E-03	1.54412E-02	1.54397E-02	8.9968	8.9776	257.39
3.45000E-03	1.58944E-02	1.58918E-02	9.1305	9.1050	257.39
3.50000E-03	1.63502E-02	1.63502E-02	9.2334	9.2323	256.94
3.55000E-03	1.68152E-02	1.68150E-02	9.3668	9.3594	256.94
3.60000E-03	1.72869E-02	1.72861E-02	9.5003	9.4864	256.94
3.65000E-03	1.77652E-02	1.77636E-02	9.6338	9.6133	256.94
3.70000E-03	1.82502E-02	1.82475E-02	9.7673	9.7400	256.94
3.75000E-03	1.87376E-02	1.87376E-02	9.8677	9.8667	256.49
3.80000E-03	1.92341E-02	1.92341E-02	10.001	9.9931	256.49
3.85000E-03	1.97377E-02	1.97369E-02	10.134	10.119	256.49
3.90000E-03	2.02478E-02	2.02461E-02	10.267	10.246	256.49
3.95000E-03	2.07645E-02	2.07615E-02	10.401	10.372	256.49
4.00000E-03	2.12832E-02	2.12832E-02	10.499	10.498	256.04
4.05000E-03	2.18115E-02	2.18113E-02	10.632	10.624	256.04
4.10000E-03	2.23464E-02	2.23456E-02	10.765	10.749	256.04
4.15000E-03	2.28880E-02	2.28862E-02	10.898	10.875	256.04
4.20000E-03	2.34362E-02	2.34331E-02	11.031	11.000	256.04
4.25000E-03	2.39862E-02	2.39862E-02	11.127	11.126	255.59
4.30000E-03	2.45459E-02	2.45457E-02	11.260	11.251	255.59
4.35000E-03	2.51122E-02	2.51113E-02	11.392	11.376	255.59
4.40000E-03	2.56851E-02	2.56832E-02	11.525	11.501	255.59
4.45000E-03	2.62647E-02	2.62614E-02	11.658	11.626	255.59
4.50000E-03	2.68458E-02	2.68458E-02	11.751	11.750	255.14
4.55000E-03	2.74367E-02	2.74364E-02	11.884	11.875	255.14
4.60000E-03	2.80342E-02	2.80333E-02	12.017	11.999	255.14
4.65000E-03	2.86384E-02	2.86364E-02	12.149	12.124	255.14
4.70000E-03	2.92491E-02	2.92457E-02	12.282	12.248	255.14
4.75000E-03	2.98612E-02	2.98612E-02	12.373	12.372	254.70
4.80000E-03	3.04831E-02	3.04828E-02	12.505	12.496	254.70
4.85000E-03	3.11117E-02	3.11107E-02	12.638	12.620	254.70
4.90000E-03	3.17469E-02	3.17448E-02	12.770	12.743	254.70
4.95000E-03	3.23887E-02	3.23850E-02	12.902	12.867	254.70
5.00000E-03	3.30315E-02	3.30315E-02	12.991	12.990	254.25
5.05000E-03	3.36843E-02	3.36840E-02	13.123	13.113	254.25
5.10000E-03	3.43438E-02	3.43428E-02	13.255	13.237	254.25
5.15000E-03	3.50099E-02	3.50077E-02	13.388	13.360	254.25
5.20000E-03	3.56825E-02	3.56788E-02	13.520	13.482	254.25
5.25000E-03	3.63559E-02	3.63559E-02	13.606	13.605	253.81
5.30000E-03	3.70396E-02	3.70393E-02	13.738	13.728	253.81
5.35000E-03	3.77298E-02	3.77287E-02	13.870	13.850	253.81
5.40000E-03	3.84266E-02	3.84243E-02	14.002	13.973	253.81
5.45000E-03	3.91300E-02	3.91260E-02	14.134	14.095	253.81
5.50000E-03	3.98338E-02	3.98338E-02	14.218	14.217	253.36
5.55000E-03	4.05480E-02	4.05477E-02	14.350	14.339	253.36
5.60000E-03	4.12688E-02	4.12677E-02	14.482	14.461	253.36

5.65000E-03	4.19962E-02	4.19938E-02	14.613	14.583	253.36
5.70000E-03	4.27301E-02	4.27260E-02	14.745	14.705	253.36
5.75000E-03	4.34620E-02	4.34620E-02	14.745	14.689	253.36
5.80000E-03	4.41982E-02	4.41950E-02	14.745	14.674	253.36
5.85000E-03	4.49321E-02	4.49249E-02	14.613	14.523	253.36
5.90000E-03	4.56595E-02	4.56472E-02	14.482	14.371	253.36
5.95000E-03	4.63803E-02	4.63620E-02	14.356	14.226	253.36
6.00000E-03	4.70718E-02	4.70718E-02	14.402	14.211	253.36
6.05000E-03	4.77886E-02	4.77786E-02	14.271	14.060	253.36
6.10000E-03	4.84989E-02	4.84778E-02	14.139	13.909	253.36
6.15000E-03	4.92025E-02	4.91695E-02	14.007	13.758	253.36
6.20000E-03	4.98996E-02	4.98536E-02	13.876	13.607	253.36
6.25000E-03	5.05325E-02	5.05325E-02	13.545	13.594	253.36
6.30000E-03	5.12131E-02	5.12153E-02	13.677	13.716	253.36
6.35000E-03	5.18980E-02	5.19019E-02	13.677	13.702	253.36
6.40000E-03	5.25785E-02	5.25832E-02	13.545	13.551	253.36
6.45000E-03	5.32525E-02	5.32570E-02	13.413	13.401	253.36
6.50000E-03	5.39256E-02	5.39256E-02	13.437	13.387	253.36
6.55000E-03	5.45941E-02	5.45912E-02	13.305	13.237	253.36
6.60000E-03	5.52561E-02	5.52492E-02	13.173	13.087	253.36
6.65000E-03	5.59114E-02	5.58998E-02	13.042	12.936	253.36
6.70000E-03	5.65602E-02	5.65429E-02	12.954	12.832	253.36
6.75000E-03	5.71807E-02	5.71807E-02	12.745	12.682	253.36
6.80000E-03	5.78191E-02	5.78157E-02	12.745	12.669	253.36
6.85000E-03	5.84537E-02	5.84461E-02	12.684	12.593	253.36
6.90000E-03	5.90846E-02	5.90720E-02	12.552	12.443	253.36
6.95000E-03	5.97102E-02	5.96917E-02	12.515	12.392	253.36
7.00000E-03	6.03075E-02	6.03075E-02	12.250	12.242	253.36
7.05000E-03	6.09212E-02	6.09205E-02	12.250	12.230	253.36
7.10000E-03	6.15332E-02	6.15311E-02	12.218	12.184	253.36
7.15000E-03	6.21408E-02	6.21366E-02	12.086	12.034	253.36
7.20000E-03	6.27418E-02	6.27346E-02	11.999	11.931	253.36
7.25000E-03	6.33274E-02	6.33274E-02	11.865	11.782	253.36
7.30000E-03	6.39217E-02	6.39173E-02	11.865	11.770	253.36
7.35000E-03	6.45117E-02	6.45021E-02	11.734	11.621	253.36
7.40000E-03	6.50951E-02	6.50794E-02	11.602	11.473	253.36
7.45000E-03	6.56719E-02	6.56494E-02	11.514	11.370	253.36
7.50000E-03	6.62141E-02	6.62141E-02	11.281	11.267	253.36
7.55000E-03	6.67793E-02	6.67784E-02	11.281	11.255	253.36
7.60000E-03	6.73444E-02	6.73420E-02	11.281	11.244	253.36
7.65000E-03	6.79052E-02	6.79005E-02	11.149	11.096	253.36
7.70000E-03	6.84594E-02	6.84516E-02	11.018	10.948	253.36
7.75000E-03	6.89975E-02	6.89975E-02	10.927	10.937	253.36
7.80000E-03	6.95428E-02	6.95429E-02	10.927	10.925	253.36
7.85000E-03	7.00858E-02	7.00855E-02	10.795	10.777	253.36
7.90000E-03	7.06223E-02	7.06207E-02	10.663	10.630	253.36
7.95000E-03	7.11521E-02	7.11485E-02	10.532	10.482	253.36
8.00000E-03	7.16712E-02	7.16712E-02	10.475	10.471	253.36

8.05000E-03	7.21916E-02	7.21911E-02	10.343	10.324	253.36
8.10000E-03	7.27077E-02	7.27059E-02	10.343	10.314	253.36
8.15000E-03	7.32216E-02	7.32179E-02	10.212	10.166	253.36
8.20000E-03	7.37289E-02	7.37225E-02	10.080	10.019	253.36
8.25000E-03	7.42198E-02	7.42198E-02	9.9444	9.8798	253.36
8.30000E-03	7.47181E-02	7.47147E-02	9.9444	9.8696	253.36
8.35000E-03	7.52120E-02	7.52045E-02	9.8127	9.7229	253.36
8.40000E-03	7.56994E-02	7.56869E-02	9.6810	9.5763	253.36
8.45000E-03	7.61801E-02	7.61621E-02	9.5932	9.4754	253.36
8.50000E-03	7.66322E-02	7.66322E-02	9.3773	9.3746	253.36
8.55000E-03	7.71022E-02	7.71018E-02	9.3773	9.3649	253.36
8.60000E-03	7.75707E-02	7.75695E-02	9.3479	9.3248	253.36
8.65000E-03	7.80343E-02	7.80320E-02	9.2162	9.1786	253.36
8.70000E-03	7.84923E-02	7.84873E-02	9.1284	9.0781	253.36
8.75000E-03	7.89376E-02	7.89376E-02	9.0073	8.9321	253.36
8.80000E-03	7.93868E-02	7.93828E-02	9.0073	8.9230	253.36
8.85000E-03	7.98339E-02	7.98253E-02	8.8756	8.7772	253.36
8.90000E-03	8.02744E-02	8.02605E-02	8.7440	8.6316	253.36
8.95000E-03	8.07083E-02	8.06885E-02	8.6123	8.4861	253.36
9.00000E-03	8.11114E-02	8.11114E-02	8.4324	8.4774	253.36
9.05000E-03	8.15341E-02	8.15362E-02	8.4324	8.4687	253.36
9.10000E-03	8.19547E-02	8.19583E-02	8.4324	8.4600	253.36
9.15000E-03	8.23730E-02	8.23777E-02	8.3007	8.3147	253.36
9.20000E-03	8.27847E-02	8.27898E-02	8.1690	8.1695	253.36
9.25000E-03	8.31946E-02	8.31946E-02	8.1514	8.0245	253.36
9.30000E-03	8.35989E-02	8.35922E-02	8.0197	7.8796	253.36
9.35000E-03	8.39966E-02	8.39826E-02	7.8881	7.7349	253.36
9.40000E-03	8.43877E-02	8.43657E-02	7.7564	7.5904	253.36
9.45000E-03	8.47744E-02	8.47439E-02	7.7564	7.5826	253.36
9.50000E-03	8.51194E-02	8.51194E-02	7.3824	7.4382	253.36
9.55000E-03	8.54918E-02	8.54945E-02	7.4702	7.5210	253.36
9.60000E-03	8.58642E-02	8.58693E-02	7.4702	7.5139	253.36
9.65000E-03	8.62344E-02	8.62414E-02	7.3385	7.3695	253.36
9.70000E-03	8.65981E-02	8.66062E-02	7.2068	7.2254	253.36
9.75000E-03	8.69639E-02	8.69639E-02	7.2174	7.1269	253.36
9.80000E-03	8.73215E-02	8.73167E-02	7.0857	6.9829	253.36
9.85000E-03	8.76725E-02	8.76622E-02	6.9541	6.8391	253.36
9.90000E-03	8.80169E-02	8.80006E-02	6.8224	6.6955	253.36
9.95000E-03	8.83547E-02	8.83318E-02	6.6907	6.5520	253.36
1.00000E-02	8.86580E-02	8.86580E-02	6.4758	6.5453	253.36
1.00500E-02	8.89851E-02	8.89886E-02	6.5636	6.6296	253.36
1.01000E-02	8.93100E-02	8.93165E-02	6.4319	6.4862	253.36
1.01500E-02	8.96283E-02	8.96372E-02	6.3442	6.3885	253.36
1.02000E-02	8.99422E-02	8.99530E-02	6.2564	6.2909	253.36
1.02500E-02	9.02649E-02	9.02640E-02	6.2474	6.1478	253.36
1.03000E-02	9.05731E-02	9.05678E-02	6.1157	6.0048	253.36
1.03500E-02	9.08756E-02	9.08645E-02	5.9840	5.8620	253.36
1.04000E-02	9.11715E-02	9.11540E-02	5.8523	5.7194	253.36

1.04500E-02	9.14608E-02	9.14364E-02	5.7645	5.6225	253.36
1.05000E-02	9.17140E-02	9.17140E-02	5.4861	5.4801	253.36
1.05500E-02	9.19917E-02	9.19913E-02	5.5759	5.5655	253.36
1.06000E-02	9.22672E-02	9.22660E-02	5.4881	5.4687	253.36
1.06500E-02	9.25383E-02	9.25359E-02	5.3564	5.3265	253.36
1.07000E-02	9.28050E-02	9.28009E-02	5.3564	5.3210	253.36
1.07500E-02	9.30634E-02	9.30634E-02	5.1966	5.1790	253.36
1.08000E-02	9.33199E-02	9.33188E-02	5.0835	5.0563	253.36
1.08500E-02	9.35712E-02	9.35684E-02	4.9658	4.9291	253.36
1.09000E-02	9.38206E-02	9.38159E-02	4.9658	4.9240	253.36
1.09500E-02	9.40656E-02	9.40586E-02	4.8342	4.7823	253.36
1.10000E-02	9.42941E-02	9.42941E-02	4.6856	4.6408	253.36
1.10500E-02	9.45273E-02	9.45249E-02	4.6856	4.6361	253.36
1.11000E-02	9.47583E-02	9.47532E-02	4.5539	4.4947	253.36
1.11500E-02	9.49827E-02	9.49744E-02	4.4222	4.3534	253.36
1.12000E-02	9.52005E-02	9.51685E-02	4.3344	4.2579	253.36
1.12500E-02	9.53979E-02	9.53979E-02	4.1611	4.1169	253.36
1.13000E-02	9.56071E-02	9.56048E-02	4.1611	4.1127	253.36
1.13500E-02	9.58118E-02	9.58069E-02	4.0294	3.9718	253.36
1.14000E-02	9.60100E-02	9.60020E-02	3.9416	3.8767	253.36
1.14500E-02	9.62038E-02	9.61923E-02	3.8539	3.7816	263.36
1.15000E-02	9.63779E-02	9.63779E-02	3.6722	3.6504	253.36
1.15500E-02	9.65626E-02	9.65614E-02	3.6722	3.6466	253.36
1.16000E-02	9.67429E-02	9.67402E-02	3.5405	3.5062	253.36
1.16500E-02	9.69188E-02	9.69143E-02	3.5405	3.5026	253.36
1.17000E-02	9.70925E-02	9.70859E-02	3.4088	3.3624	253.36
1.17500E-02	9.72506E-02	9.72506E-02	3.2764	3.2223	253.36
1.18000E-02	9.74111E-02	9.74082E-02	3.1886	3.1280	253.36
1.18500E-02	9.75672E-02	9.75611E-02	3.0690	3.0008	253.36
1.19000E-02	9.77196E-02	9.77099E-02	3.0690	2.9977	253.36
1.19500E-02	9.78697E-02	9.78563E-02	2.9373	2.8580	253.36
1.20000E-02	9.79957E-02	9.79957E-02	2.7254	2.7184	253.36
1.20500E-02	9.81331E-02	9.81327E-02	2.7254	2.7156	253.36
1.21000E-02	9.82661E-02	9.82650E-02	2.5938	2.5762	253.36
1.21500E-02	9.83947E-02	9.83926E-02	2.5938	2.5736	253.36
1.22000E-02	9.85210E-02	9.85178E-02	2.4621	2.4343	253.36
1.22500E-02	9.86360E-02	9.86360E-02	2.3517	2.2952	263.36
1.23000E-02	9.87503E-02	9.87473E-02	2.2200	2.1562	253.36
1.23500E-02	9.88580E-02	9.88517E-02	2.1322	2.0630	253.36
1.24000E-02	9.89614E-02	9.89513E-02	2.0005	1.9242	253.36
1.24500E-02	9.90603E-02	9.90464E-02	2.0005	1.9223	253.36
1.25000E-02	9.91390E-02	9.91390E-02	1.7372	1.7837	253.36
1.25500E-02	9.92270E-02	9.92293E-02	1.7372	1.7818	253.36
1.26000E-02	9.93127E-02	9.93172E-02	1.7372	1.7800	253.36
1.26500E-02	9.93961E-02	9.94027E-02	1.6055	1.6416	253.36
1.27000E-02	9.94731E-02	9.94813E-02	1.4738	1.5032	253.36
1.27500E-02	9.95531E-02	9.95531E-02	1.4783	1.3651	253.36
1.28000E-02	9.96237E-02	9.96179E-02	1.3467	1.2271	253.36

```
1.28500E-02  9.96877E-02  9.96758E-02  1.2150    1.0892    253.36
1.29000E-02  9.97452E-02  9.97268E-02  1.0833    0.95144   253.36
1.29500E-02  9.97961E-02  9.97709E-02  0.95161   0.81384   253.36
1.30000E-02  9.98082E-02  9.98092E-02  0.70256   0.67638   253.36
1.30500E-02  9.98466E-02  9.98454E-02  0.79035   0.76674   253.36
1.31000E-02  9.98828E-02  9.98803E-02  0.70256   0.67490   253.36
1.31500E-02  9.99146E-02  9.99106E-02  0.61477   0.58315   253.36
1.32000E-02  9.99421E-02  9.99363E-02  0.48309   0.44593   253.36
1.32500E-02  9.99575E-02  9.99575E-02  0.38560   0.39993   253.36
1.33000E-02  9.99778E-02  9.99786E-02  0.38560   0.39950   253.36
1.33500E-02  9.99938E-02  9.99951E-02  0.25392   0.26247   253.36
1.34000E-02  0.10000      0.10000      0.12224   0.12558   253.36
$SIMULATION HALTED:      C        =     0.100003

              >>>> DSL SIMULATION INPUT DATA <<<<
       STOP
```

APPENDIX C

DSL/VS Program for the Current Source Drive System of Chapter 5

DSL/VS Simulation Data for the Step Response of Figure 5.6 (1 Track Move)

```
TITLE   APP C(1)
TITLE   SIMULATION PROGRAM OF THE CURRENT SOURCE DRIVE SYSTEM
PARAM   K1=0.70.K2=10000.0.KM=300.00.VSAT=10.00.K=1.0.T=0.00025
PARAM   REF=0.0010
******  CURRENT SOURCE DRIVE PARAMETERS *******
PARAM   RA=1.012.L=0.0002862.RSENS=0.10.KV=0.0752.KTOJ=275.2
PARAM   KA=50.00.VSATA=50.0
INTGER  N.FLAG
INITIAL
            N=0
            FLAG=0
            RTOT=RA+RSENS
            KFB=1./RSENS
            PP2=1./20.555
            PP3=1./(RTOT/L)
            A=SQRT(2.*KM*VSAT)
            XDOT=0.0
            CDOT=0.0
DERIVATIVE
         R=REF*STEP(0.0)
         E=R-C
NOSORT
         IF(E.LT.0.0)XDOT=-A*K1*SQRT(ABS(E))
         IF(E.GE.0.0)XDOT=A*K1*SQRT(E)
SORT
         XDOTE=XDOT-KCDOT
         KCDOT=CDOT*K
         V=LIMIT(-VSAT.VSAT.K2*XDOTE)
NOSORT
         IF(FLAG.EQ.1)GO TO 5
         IF(V.LT.VSAT.AND.TIME.GT.0.00005)FLAG=1
         NSW=N
5        CONTINUE
SORT
         CDDOT=KM*V
         CDOT=INTGRL(0.0.CDDOT)
         C=INTGRL(0.0.CDOT)
         VS=LIMIT(-VSAT.VSAT.K2*XDOTE)
*********   CURRENT SOURCE DRIVE AMPLIFIER AND MOTOR *************
         VFB=KFB*I*RSENS
         VIA=VS-VFB
         VINA=LIMIT(-VSATA.VSATA.VIA*KA)
         VA=VINA-KV*CSDOT
         IA=(1./RTOT)*VA
         I=REALPL(0.0.PP3.IA)
         CSDDOT=KTOJ*I
         CSDOT=INTGRL(0.0.CSDDOT)
         CS=INTGRL(0.0.CSDOT)
```

```
SAMPLE
NOSORT
      IF(N.EQ.0)GO TO 20
      C=CS
      KS=ABS(2.*CS/(VS*((N*T)**2)))
      IF(FLAG.EQ.0)KM=KS
      IF(N.GE.2)CSDOTL=(CS-CS2L)/(2.*T)
      IF (FLAG.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
      IF(N.EQ.NSW.AND.FLAG.EQ.1)GO TO 20
      IF (FLAG.EQ.1)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
20    N=N+1
      CSDOTL=CDOT
      CS2L=CSLAST
      CSLAST=CS
SORT
TERMINAL
FINISH   C=0.0010
METHOD RKS
CONTRL  FINTIM=0.05000.DELT=0.00005.DELS=.00025
PRINT  0.00005.CS.CDOT.CSDOT.VINA.I
SAVE (G1)0.00005.TIME.C.XDOT.CDOT.CSDOT
SAVE (G3)0.00005.TIME.C.CS.REF
*RAPH(G1/G1.DE=TEK618.PO=0..5) C(LE=7.5.UN='INCHES')....
*       XDOT(LE=12.NI=12.LO=-2..SC=2.0.AX=OMIT)....
*       CDOT(LE=12.NI=12.LO=-2.0.SC=2.0.UN='INCHES/SEC')....
*       CSDOT(LE=12.NI=12.LO=-2.0.PO=7.5.SC=2.0.UN='INCHES/SEC')
*RAPH(G3/G3.DE=TEK618.PO=0..5) TIME(LE=7.5.NI=6.SC=0.0025....
*       UN=SECONDS)....
*       C(LE=12.NI=12.LO=0.SC=0.02.UN='INCHES')....
*       CS(LE=12.NI=12.LO=0.PO=7.5.SC=0.02.UN='INCHES')....
*       REF(LE=12.NI=12.LO=0.SC=0.02.AX=OMIT)
GRAPH(G1/G1.DE=TEK618.PO=0..5) C(LE=7.0.SC=0.0001.NI=11.UN='INCHES')....
        XDOT(LE=12.NI=12.LO=-.20.SC=.20.AX=OMIT)....
        CDOT(LE=12.NI=12.LO=-.20.SC=.20.UN='INCHES/SEC')....
        CSDOT(LE=12.NI=12.LO=-.20.PO=7.0.SC=.20.UN='INCHES/SEC')
GRAPH(G3/G3.DE=TEK618.PO=0..5) TIME(LE=7.0.NI=15.SC=0.0001....
        UN=SECONDS)....
        C(LE=12.NI=12.LO=0.SC=0.0002.UN='INCHES')....
        CS(LE=12.NI=12.LO=0.PO=7.0.SC=0.0002.UN='INCHES')....
        REF(LE=12.NI=12.LO=0.SC=0.0002.AX=OMIT)
LABEL (G1)  PHASE PLANE
LABEL (G1)  XDOT.CDOT.CSDOT VS C
LABEL (G3)  STEP RESPONSE
LABEL (G3)  C.CS VS TIME
END
STOP
```

```
               >>>> DSL SIMULATION INPUT DATA <<<<
TITLE   APP C(2)
TITLE   DATAM FOR THE CURRENT SOURCE DRIVE SYSTEM(ITRACK MOVE)
PARAM   K1=0.70,K2=10000.0,KM=300.00,VSAT=10.00,K=1.0,T=0.00025
PARAM   REF=0.0010
PARAM   RA=1.012,L=0.0002862,RSENS=0.10,KV=0.0752,KTOJ=275.2
PARAM   KA=50.00,VSATA=50.0
INTGER N,FLAG
FINISH    C=0.0010
METHOD RKS
CONTRL   FINTIM=0.05000,DELT=0.00005,DELS=.00025
PRINT   0.00005,CS,CDOT,CSDOT,VINA,I
SAVE  (G1)0.00005,TIME,C,XDOT,CDOT,CSDOT
SAVE  (G3)0.00005,TIME,C,CS,REF
GRAPH(G1/G1,DE=TEK618,PO=0.,5) C(LE=7.0,SC=0.0001,NI=11....          =====
          XDOT(LE=12,NI=12,LO=-.20,SC=.20,AX=OMIT),...
          CDOT(LF=12,NI=12,LO=-.20,SC=.20,UN='INCHES/SEC')....
          CSDOT(LE=12,NI=12,LO=-.20,PO=7.0,SC=.20,UN='INCHES/SEC' )
GRAPH(G3/G3,DE=TEK618,PO=0.,5) TIME(LE=7.0,NI=15,SC=0.0001,...
          UN=SECONDS)....
          C(LE=12,NI=12,LO=0,SC=0.0002,UN='INCHES')....
          CS(LE=12,NI=12,LO=0,PO=7.0,SC=0.0002,UN='INCHES')....
          REF(LE=12,NI=12,LO=0,SC=0.0002,AX=OMIT)
LABEL (G1)   PHASE PLANE
LABEL (G1)   XDOT,CDOT,CSDOT VS C
LABEL (G3)   STEP RESPONSE
LABEL (G3)   C,CS VS TIME
END
****   RKS      INTEGRATION METHOD USED ***
         *** DSL OUTPUT LISTING, GROUP  1
          APP C(1)
          SIMULATION PROGRAM OF THE CURRENT SOURCE DRIVE SYSTEM
```

TIME	CS	CDOT	CSDOT	VINA	I
0.00000E+00	0.00000E+00	0.00000E+00	0.00000E+00	50.000	0.0000
5.00000E-05	9.54800E-07	0.15000	5.63872E-02	50.000	7.9388
1.00000E-04	6.96954E-06	0.30000	0.18709	10.912	9.7818
1.50000E-04	1.96879E-05	0.45000	0.32169	10.913	9.7817
2.00000E-04	3.91373E-05	0.60000	0.45629	10.918	9.7816
2.50000E-04	6.53165E-05	0.52253	0.59088	10.921	9.7816
3.00000E-04	9.82255E-05	0.62704	0.72548	10.930	9.7814
3.50000E-04	1.37864E-04	0.73155	0.86007	10.941	9.7812
4.00000E-04	1.84232E-04	0.83605	0.99465	10.950	9.7810
4.50000E-04	2.37330E-04	0.94056	1.1292	10.962	9.7808
5.00000E-04	2.97156E-04	1.2604	1.2638	10.970	9.7806
5.50000E-04	3.63711E-04	1.3690	1.3981	-50.000	9.1281
6.00000E-04	4.39607E-04	1.2954	1.4558	-50.000	-0.43983
6.50000E-04	5.07330E-04	1.2219	1.3992	-7.5676	-6.0341
7.00000E-04	5.75210E-04	1.1484	1.3160	-6.5674	-6.0544

7.50000E-04	6.38925E-04	1.0748	1.2327	-50.000	-6.0544
8.00000E-04	6.97746E-04	0.95600	1.1117	50.000	-9.7534
8.50000E-04	7.50823E-04	0.88221	1.0184	-6.4799	-6.0563
9.00000E-04	7.99661E-04	0.80866	0.93510	-6.5724	-6.0552
9.50000E-04	8.44333E-04	0.73510	0.85178	-6.5999	-6.0555
1.00000E-03	8.84839E-04	0.79194	0.76846	-50.000	-6.0559
1.05000E-03	9.20450E-04	0.67308	0.64749	-11.023	-9.7795
1.10000E-03	9.49459E-04	0.55422	0.51287	-10.951	-9.7810
1.15000E-03	9.71737E-04	0.43536	0.37826	-10.914	-9.7817
1.20000E-03	9.87285E-04	0.31649	0.24364	-10.897	-9.7821
1.25000E-03	9.96102E-04	0.17575	0.10903	-10.891	-9.7822
1.30000E-03	9.98188E-04	5.68852E-02	-2.55737E-02	-10.891	-9.7822

*SIMULATION HALTED: C = 1.001918E-03

APPENDIX D

DSL/VS Program, Seek and Track Follow Modes, with Compensation Closed Around the Servo Motor (2nd Order Motor)

DSL/VS Program, Seek and Track Follow Modes, with Compensation Filter Placed in the Model (2nd Order Motor)

DSL/VS Program, Seek and Track Follow Modes, with Model Tach Feedback Compensation (2nd Order Motor)

```
TITLE   APP D(1)
TITLE   COMPENSATION CLOSED AROUND THE SERVO (2ND ORDER MOTOR)
PARAM   K1=0.80,K2=10000.0,KM=300.00,VSAT=10.,K=1.0,T=0.00025
PARAM   REF=0.0010
PARAM   Z=1000.,P=50000.,KC=50.0
INTGER  N,FLAG,FLAGC
INITIAL
          N=0
          FLAG=0
          FLAGC=0
          PP2=1./20.555
          PP3=1./3536.
          ZZ=1./Z
          PP=1./P
          A=SQRT(2.*KM*VSAT)
          XDOT=0.0
DYNAMIC
        IF(FLAGC.EQ.1)GO TO 5
        IF(CDOT.LT.1.000.AND.TIME.GT.0.0005)FLAGC=1
*       IF(E.LT.0.000.AND.TIME.GT.0.0005)FLAGC=1
        IF(FLAGC.EQ.1)K=0.0
        IF(FLAG.EQ.1)GO TO 5
        IF(V.LT.VSAT.AND.TIME.GT.0.0005)FLAG=1
        NSW=N
5       CONTINUE
DERIVATIVE
        R=REF*STEP(0.0)
        E=R-C
NOSORT
        IF(E.GE.0.0)XDOT=A*K1*SQRT(E)
        IF(E.LT.0.0)XDOT=-A*K1*SQRT(ABS(E))
SORT
        XDOTE=XDOT-KCDOT
        KCDOT=CDOT*K
        V=LIMIT(-VSAT,VSAT,K2*XDOTE)
        CDDOT=KM*V
        CDOT=INTGRL(0.0,CDDOT)
        C=INTGRL(0.0,CDOT)
NOSORT
        IF(FLAGC.EQ.0)ES=0.0
        IF(FLAGC.EQ.1)ES=R-CSSAMP
SORT
        ES1=KC*ES
        ES2=ES1*(Z-P)
        ES3=ES2-P*ES4
        ES4=INTGRL(0.0,ES3)
        GC=ES1+ES4
NOSORT
```

```
        IF(FLAGC.EQ.0)VS=LIMIT(-VSAT.VSAT.K2*XDOTE)
        IF(FLAGC.EQ.1)VS=LIMIT(-VSAT.VSAT.K2*GC)
SORT
        CSDDOT=13.3*VS
        CSDOT=REALPL(0.0.PP2.CSDDOT)
        CS=INTGRL(0.0.CSDOT)
SAMPLE
NOSORT
        IF(N.EQ.0)GO TO 20
        C=CS
        KS=ABS(2.*CS/(VS*((N*T)**2)))
        IF(FLAG.EQ.0)KM=KS
        IF(N.GE.2)CSDOTL=(CS-CS2L)/(2.*T)
        IF (FLAG.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
        IF(N.EQ.NSW.AND.FLAG.EQ.1)GO TO 20
        IF (FLAG.EQ.1)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
        IF(FLAGC.EQ.1)CDOT=(CS-CSLAST)/T
20      N=N+1
        CSDOTL=CDOT
        CS2L=CSLAST
        CSLAST=CS
        CSSAMP=CS
SORT
TERMINAL
METHOD RKSFX
CONTRL  FINTIM=0.00500.DELT=0.00005.DELS=.00025
PRINT  0.00005.C.CS.CDOT.CSDOT.E.ES.GC.VS.FLAG.FLAGC
SAVE (G1)0.00005.TIME.C.XDOT.CDOT.CSDOT
SAVE (G3)0.00005.TIME.C.CS.REF
*RAPH(G1/G1.DE=TEK618.PO=0..5) C(LE=7.5.UN='INCHES')....
*          XDOT(LE=12.NI=12.LO=0.SC=2.0.AX=OMIT)....
*          CDOT(LE=12.NI=12.LO=0.SC=2.0.UN='INCHES/SEC')....
*          CSDOT(LE=12.NI=12.LO=0.PO=7.5.SC=2.0.UN='INCHES/SEC')
*RAPH(G3/G3.DE=TEK618.PO=0..5) TIME(LE=7.5.NI=6.SC=0.0025....
*          UN=SECONDS)....
*          C(LE=12.NI=12.LO=0.SC=0.02.UN='INCHES')....
*          CS(LE=12.NI=12.LO=0.PO=7.5.SC=0.02.UN='INCHES')....
*          REF(LE=12.NI=12.LO=0.SC=0.02.AX=OMIT)
GRAPH(G1/G1.DE=TEK618.PO=0..5) C(LE=7.0.SC=0.0001.NI=11.UN='INCHES')....
          XDOT(LE=12.NI=12.LO=-.20.SC=.20.AX=OMIT)....
          CDOT(LE=12.NI=12.LO=-.20.SC=.20.UN='INCHES/SEC')....
          CSDOT(LE=12.NI=12.LO=-.20.PO=7.0.SC=.20.UN='INCHES/SEC')
GRAPH(G3/G3.DE=TEK618.PO=0..5) TIME(LE=8.0.NI=10.SC=0.0005....
          UN=SECONDS)....
          CS(LE=12.NI=12.LO=0.PO=0.0.SC=0.0002.UN='INCHES')....
          REF(LE=12.NI=12.LO=0.SC=0.0002.AX=OMIT)
LABEL (G1)   PHASE PLANE
LABEL (G1)   XDOT.CDOT.CSDOT VS C
```

```
LABEL (G3)  STEP RESPONSE
LABEL (G3)  C.CS VS TIME
END
STOP
```

```
TITLE   APP D(2)
TITLE   COMPENSATION FILTER IN THE MODEL
PARAM   K1=0.80,K2=10000.0,KM=300.00,VSAT=10.,K=1.0,T=0.00025
PARAM   REF=0.0010
PARAM   Z=1000.,P=50000.
PARAM   KC=50.
INTGER  N,FLAG,FLAGC
INITIAL
          N=0
          FLAG=0
          FLAGC=0
          PP2=1./20.555
          PP3=1./3536.
          ZZ=1./Z
          PP=1./P
          A=SQRT(2.*KM*VSAT)
          XDOT=0.0
DYNAMIC
        IF(FLAGC.EQ.1)GO TO 5
*       IF(CDOT.LT.0.000.AND.TIME.GT.0.0005)FLAGC=1
        IF(E.LT.0.000.AND.TIME.GT.0.0005)FLAGC=1
        IF(FLAGC.EQ.1)K=0.0
        IF(FLAG.EQ.1)GO TO 5
        IF(V.LT.VSAT.AND.TIME.GT.0.0005)FLAG=1
        NSW=N
5       CONTINUE
DERIVATIVE
        R=REF*STEP(0.0)
        E=R-C
NOSORT
        IF(FLAGC.EQ.0)ES=0.0
        IF(FLAGC.EQ.1)ES=R-C
        ES1=KC*ES
        ES2=ES1*(Z-P)
        ES3=ES2-P*ES4
        ES4=INTGRL(0.0,ES3)
        GC=ES1+ES4
        IF(E.GE.0.0.AND.FLAGC.EQ.0)XDOT=A*K1*SQRT(E)
        IF(E.LT.0.0.AND.FLAGC.EQ.0)XDOT=-A*K1*SQRT(ABS(E))
        IF(FLAGC.EQ.1)XDOT=GC
SORT
        XDOTE=XDOT-KCDOT
        KCDOT=CDOT*K
        V=LIMIT(-VSAT,VSAT,K2*XDOTE)
        CDDOT=KM*V
        CDOT=INTGRL(0.0,CDDOT)
        C=INTGRL(0.0,CDOT)
        VS=LIMIT(-VSAT,VSAT,K2*XDOTE)
```

```
      CSDDOT=13.3*VS
      CSDOT=REALPL(0.0,PP2,CSDDOT)
      CS=INTGRL(0.0,CSDOT)
SAMPLE
NOSORT
      IF(N.EQ.0)GO TO 20
      C=CS
      KS=ABS(2.*CS/(VS*((N*T)**2)))
      IF(FLAG.EQ.0)KM=KS
      IF(N.GE.2)CSDOTL=(CS-CS2L)/(2.*T)
      IF (FLAG.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
      IF(N.EQ.NSW.AND.FLAG.EQ.1)GO TO 20
      IF (FLAG.EQ.1.AND.FLAGC.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
      IF(FLAGC.EQ.1)CDOT=(CS-CSLAST)/T
20    N=N+1
      CSDOTL=CDOT
      CS2L=CSLAST
      CSLAST=CS
      CSSAMP=CS
SORT
TERMINAL
METHOD RKSFX
CONTRL  FINTIM=0.00500,DELT=0.00005,DELS=.00050
PRINT  0.00005,C,CS,CDOT,CSDOT,E,ES,GC,VS,FLAG,FLAGC
SAVE (G1)0.00005,TIME,C,XDOT,CDOT,CSDOT
SAVE (G3)0.00005,TIME,C,CS,REF
*RAPH(G1/G1,DE=TEK618,PO=0,.5) C(LE=7.5,UN='INCHES')....
*       XDOT(LE=12,NI=12,LO=0,SC=2.0,AX=OMIT)....
*       CDOT(LE=12,NI=12,LO=0,SC=2.0,UN='INCHES/SEC')....
*       CSDOT(LE=12,NI=12,LO=0,PO=7.5,SC=2.0,UN='INCHES/SEC')
*RAPH(G3/G3,DE=TEK618,PO=0,.5) TIME(LE=7.5,NI=6,SC=0.0025....
*       UN=SECONDS)....
*       C(LE=12,NI=12,LO=0,SC=0.02,UN='INCHES')....
*       CS(LE=12,NI=12,LO=0,PO=7.5,SC=0.02,UN='INCHES')....
*       REF(LE=12,NI=12,LO=0,SC=0.02,AX=OMIT)
GRAPH(G1/G1,DE=TEK618,PO=0,.5) C(LE=7.0,SC=0.0001,NI=11,UN='INCHES')....
        XDOT(LE=12,NI=12,LO=-.20,SC=.20,AX=OMIT)....
        CDOT(LE=12,NI=12,LO=-.20,SC=.20,UN='INCHES/SEC')....
        CSDOT(LE=12,NI=12,LO=-.20,PO=7.0,SC=.20,UN='INCHES/SEC')
GRAPH(G3/G3,DE=TEK618,PO=0,.5) TIME(LE=7.0,NI=10,SC=0.0005....
        UN=SECONDS)....
        C(LE=12,NI=12,LO=0,PO=0.0,SC=0.0002,UN='INCHES')....
        CS(LE=12,NI=12,LO=0,PO=7.0,SC=0.0002,UN='INCHES')....
        REF(LE=12,NI=12,LO=0,SC=0.0002,AX=OMIT)
LABEL (G1)   PHASE PLANE
LABEL (G1)   XDOT,CDOT,CSDOT VS C
LABEL (G3)   STEP RESPONSE
LABEL (G3)   C,CS VS TIME
```

```
TITLE   APP D(3)
TITLE   MODEL TACH FEEDBACK COMPENSATION IN TRACK FOLLOW
INTGER  FLAG,FLAGC
PARAM   K1=0.80,K2=10000.0,KM=300.00,VSAT=10.,K=1.0,T=0.000250
PARAM   REF=0.0010
INITIAL
          N=0
          FLAG=0
          FLAGC=0
          PP2=1./20.555
          PP3=1./3536.
          A=SQRT(2.*KM*VSAT)
DYNAMIC
        IF(FLAGC.EQ.1)GO TO 5
*       IF(CDOT.LT.0.000.AND.TIME.GT.0.0005)FLAGC=1
        IF(E.LE.0.00000.AND.TIME.GT.0.0005)FLAGC=1
        IF(FLAGC.EQ.1)K=0.01
5       CONTINUE
        IF(FLAG.EQ.1)GO TO 10
        IF(V.LT.VSAT.AND.TIME.GT.0.0005)FLAG=1
        NSW=N
10      CONTINUE
DERIVATIVE
        R=REF*STEP(0.0)
        E=R-C
NOSORT
        IF(E.GE.0.00000.AND.FLAGC.EQ.0)XDOT=A*K1*SQRT(E)
        IF(E.LT.0.00000.AND.FLAGC.EQ.0)XDOT=-A*K1*SQRT(ABS(E))
        IF(FLAGC.EQ.1)XDOT=32.000*E
SORT
        XDOTE=XDOT-KCDOT
        KCDOT=CDOT*K
        V=LIMIT(-VSAT,VSAT,K2*XDOTE)
        CDDOT=KM*V
        CDOT=INTGRL(0.0,CDDOT)
        C=INTGRL(0.0,CDOT)
        VS=LIMIT(-VSAT,VSAT,K2*XDOTE)
        CSDDOT=13.3*VS
        CSDOT=REALPL(0.0,PP2,CSDDOT)
        CS=INTGRL(0.0,CSDOT)
SAMPLE
NOSORT
        IF(N.EQ.0)GO TO 20
        C=CS
        KS=ABS(2.*CS/(VS*((N*T)**2)))
        IF(FLAG.EQ.0)KM=KS
        IF(N.GE.2)CSDOTL=(CS-CS2L)/(2.*T)
        IF (FLAG.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
```

```
      IF(N.EQ.NSW.AND.FLAG.EQ.1)GO TO 20
      IF (FLAG.EQ.1.AND.FLAGC.EQ.0)CDOT=(2.*((CS-CSLAST)/T))-CSDOTL
      IF (FLAGC.EQ.1)CDOT=(CS-CSLAST)/T
20    N=N+1
      CSDOTL=CDOT
      CS2L=CSLAST
      CSLAST=CS
SORT
TERMINAL
METHOD RKSFX
CONTRL   FINTIM=0.00500.DELT=0.00005.DELS=.000500
PRINT  0.00005.C.CS.CDOT.CSDOT.VS.E.XDOT.XDOTE.KCDOT.FLAG.FLAGC.KM
SAVE (G1)0.00005.TIME.C.XDOT.CDOT.CSDOT
SAVE (G3)0.00005.TIME.C.CS.REF
SAVE (G2)0.00005.TIME.CSDOT
SAVE (G4)0.00005.TIME.VS
*RAPH( G1/G1.DE=TEK618.PO=0..5) C(LE=7.5.UN='INCHES')....
*         XDOT(LE=12.NI=12.LO=-2.0.SC=2.0.AX=OMIT)....
*         CDOT(LE=12.NI=12.LO=-2.0.SC=2.0.UN='INCHES/SEC')....
*         CSDOT(LE=12.NI=12.LO=-2.0.PO=7.5.SC=2.0.UN='INCHES/SEC')
*RAPH( G3/G3.DE=TEK618.PO=0..5) TIME(LE=7.5.NI=10.SC=0.0020....
*         UN=SECONDS)....
*         C(LE=12.NI=12.LO=0.SC=0.02.UN='INCHES')....
*         CS(LE=12.NI=12.LO=0.PO=7.5.SC=0.02.UN='INCHES')....
*         REF(LE=12.NI=12.LO=0.SC=0.02.AX=OMIT)
GRAPH( G1/G1.DE=TEK618.PO=0..5) C(LE=7.0.SC=0.0001.NI=11.UN='INCHES')....
          XDOT(LE=12.NI=12.LO=-.20.SC=.20.AX=OMIT)....
          CDOT(LE=12.NI=12.LO=-.20.SC=.20.UN='INCHES/SEC')....
          CSDOT(LE=12.NI=12.LO=-.20.PO=7.0.SC=.20.UN='INCHES/SEC')
GRAPH( G3/G3.DE=TEK618.PO=0..5) TIME(LE=7.0.NI=10.SC=0.0005....
          UN=SECONDS)....
          C(LE=12.NI=12.LO=0.SC=0.0002.UN='INCHES')....
          CS(LE=12.NI=12.LO=0.PO=7.0.SC=0.0002.UN='INCHES')....
          REF(LE=12.NI=12.LO=0.SC=0.0002.AX=OMIT)
*RAPH( G2/G2.DE=TEK618.PO=0..5) TIME(LE=7.0.NI=10.SC=0.0005)....
*         CSDOT(LE=12.NI=12.LO=-.20.PO=7.0.SC=.20.UN='INCHES/SEC')
*RAPH( G4/G4.DE=TEK618.PO=0..5) TIME(LE=7.0.NI=10.SC=0.0005....
*         UN=SECONDS)....
*         VS(LE=12.NI=12.LO=-10.PO=7.0.SC=2.0.UN='VOLTS')
LABEL (G1)  PHASE PLANE
LABEL (G1)  XDOT.CDOT.CSDOT VS C
LABEL (G3)  STEP RESPONSE
LABEL (G3)  C.CS VS TIME
LABEL (G2)  SERVO MOTOR VELOCITY
LABEL (G2)  CSDOT VS TIME
LABEL (G4)  SERVO MOTOR DRIVE SIGNAL
LABEL (G4)  VS VS TIME
END
```

LIST OF REFERENCES

1. Thaler, G. J. Nonlinear Feedback Controls Theory, and Methods for Analysis and Design, unpublished notes for ECE 4412, Nonlinear Controls, Naval Postgraduate School, 1985

2. Thaler, G. J. and Stein, W. A. "Transfer Function and Parameter Evaluation for D-C Servomotors," Applications and Industry, pp 410-417, January 1956

3. Philips, C. L. and Nagle, H. T. Digital Control System Analysis and Design, Prentice Hall, 1984

4. Tzafestas, S. G., editor, Microprocessors in Signal Processing, Measurement and Control, D. Reidel Publishing Company, 1983

BIBLIOGRAPHY

International Business Machines Corporation, Dynamic Simulation Language/VS Program Description/Operations Manual, 1984

Katz, Paul, Digital Control Using Microprocessors, Prentice-Hall, 1981

Kuo, Benjamin C., Automatic Control Systems, Prentice-Hall, 1962

INITIAL DISTRIBUTION LIST

		No. of Copies
1.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5100	2
2.	Defense Techinical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
3.	Department Chairman, Code 62 Department of Electrical & Computer Engineering Naval Postgraduate School Monterey, California 93943-5100	1
4.	Professor G. J. Thaler, Code 62Tr Department of Electrical & Computer Engineering Naval Postgraduate School Monterey, California 93943-5100	5
5.	Professor H. A.Titus, Code 62Ti Department of Electrical & Computer Engineering Naval Postgraduate School Monterey, California 93943-5100	1
6.	Lt. K. R. Wikstrom 13763 Whispering Meadows Jamul, California 92035	2
7.	Mr. James Toreson XEBEC 2221 Old Oakland Road San Jose, California 95131	1
8.	Mr. Tom Hickok XEBEC 2221 Old Oakland Road San Jose, California 95131	1
9.	Mr. Ron Lesti XEBEC 2221 Old Oakland Road San Jose, California 95131	1